L.2500

XELECTRON COMPUTER

Supplemento a CQelettronica n°6

spedi. in abb. post. g. III - giugno 1984

ACCESSORI PER COMPUTER

GVH
GIANNI VECCHIETTI
Via Della Beverara 39
40131 BOLOGNA - Tel. 051/370687

NON SI VENDE AI PRIVATI

IC-R 71 Ricevitore HF a copertura generale 100 KHz - 30 MHz

La nuova versione è piú che un miglioramento dell'ormai noto R70 in quanto comprende nuove funzioni e flessibilità operative piú estese. Il segnale all'atto della prima conversione è convertito a 70 MHz eliminando in tale modo le frequenze immagini e spurie. Detto stadio ha una configurazione bilanciata con l'uso di J FET, il che porta la dinamica a 105 dB!
La successiva media frequenza a 9 MHz incorpora inoltre il filtro di assorbimento, la costante AGC regolabile, un soppressore dei disturbi capace di eliminare pure l'interferenza del fastidioso radar oltre l'orizzonte sovietico nonchè quell'indispensabile controllo di banda passante con cui è possibile regolare la selettività in modo ottimale compatibilmente con la situazione in banda. La presenza del µP rende piacevole e rilassante l'uso dell'apparato: doppio VFO ad incrementi di 10 Hz oppure 50 Hz con una rotazione piú veloce del controllo di sintonia. Un apposito tasto seleziona volendo incrementi di 1 KHz. L'impostazione della frequenza può essere eseguita in modo ancora piú veloce mediante la tastiera. Le frequenze d'interesse possono inoltre essere registrate in 32 memorie alimentate in modo indipendente da un apposito elemento al litio. È possibile la ricerca entro lo spettro HF, oppure entro le memorie. In quest'ultimo caso possono essere selezionate soltanto le classi d'emissione programmate. Una frequenza precedentemente registrata potrà essere trasferita al VFO per eseguire delle variazioni addizionali, trasferita all'altro VFO se necessario, oppure nuovamente in memoria.
Un preamplificatore con un valore intrinseco di basso valore ed inseribile se la situazione lo richiede, eleva grandemente la sensibilità. L'IC-R71 inoltre è al passo con i tempi: l'apposita interfaccia IC-10 permette di collegarlo al calcolatore. Con il generatore di fonemi si otterrà l'annuncio in inglese con una voce dall'accento femminile. Un apposito telecomando a raggi infrarossi RC-11 (opzionale) permette di manovrare l'apparato anche a distanza similarmente ad un televisore. Il grafista o lo sperimentatore piú esigente potranno avvalersi dell'apposito filtro stretto da 500 Hz, oppure del campione CR-64 che debitamente termostatato, presenta caratteristiche di elevatissima stabilità.

CARATTERISTICHE TECNICHE

RICEVITORE

Configurazione: a 4 conversioni con regolazione continua della banda passante. Tre conversioni in FM. Emissioni demodulabili: A1, USB, LSB, F1, A3, F3.
Medie frequenze: 70.4515 MHz, 9.0115 MHz, 455 KHz.
Sensibilità (con il preamplificatore incluso): SSB/CW/RTTY: $< 0.15\ \mu V$ (0.1-1.6 MHz = $1\mu V$) per 10 dB S + D/D
AM: $< 0.5\mu V$ (0.1-1.6 MHz = $3\mu V$) FM*: $< 0.3\mu V$ per 12 dB SINAD (1.6 - 30 MHz)
Selettività: SSB, CW, RTTY: 2,3 KHz a −6 dB (regolabile a 500 Hz minimi) 4,2 KHz a −60 dB
CW-N, RTTY-N: 500 Hz a −6 dB 1.5 KHz a −60 dB.
AM: 6 KHz a −6 dB (regolabile a 2.7 KHz min.) 15 KHz a −50 dB
FM*: 15 KHz a −6 dB 25 KHz a −60 dB
Reiezione a spurie ed immagini: > 60 dB
Uscita audio: > di 3 W
Impedenza audio: 8Ω
*Con l'installazione dell'unità FM opzionale.

ASSISTENZA TECNICA
S.A.T. - v. Washington, 1 Milano - tel. 432704
Centri autorizzati:
A.R.T.E. - v. Mazzini, 53 Firenze - tel. 243251
RTX Radio Service - v. Concordia, 15 Saronno tel. 9624543
e presso tutti i rivenditori Marcucci S.p.A.

EDITORE
edizioni CD s.n.c.

DIRETTORE RESPONSABILE
Giorgio Totti

REDAZIONE, AMMINISTRAZIONE, ABBONAMENTI, PUBBLICITÀ
40121 Bologna - via Cesare Boldrini 22
(051) 552706-551202
Registrazione tribunale di Bologna n. 3330 del 4/3/1968. Diritti riproduzioni traduzioni riservati a termine di legge. Iscritta al Reg. Naz. Stampa di cui alla legge n. 416 art. 11 del 5/8/81 col n. 00653 vol. 7 foglio 417 in data 18/12/82. Spedizione in abbonamento postale - gruppo III
Pubblicità inferiore al 70%

DISTRIBUZIONE PER L'ITALIA
SODIP - 20125 Milano - via Zuretti 25
Tel. (02) 67709

DISTRIBUZIONE PER L'ESTERO
Messaggerie Internazionali
via Calabria 23
20090 Fizzonasco di Pieve E. - Milano

Cambio indirizzo L. 1.000 in francobolli
Manoscritti, disegni, fotografie, anche se non pubblicati, non si restituiscono.

ABBONAMENTO
(CQ elettronica + XÉLECTRON)
Italia annuo L. 36.000 (nuovi)
L. 35.000 (rinnovi)

ABBONAMENTO ESTERO L. 43.000
Mandat de Poste International
Postanweisung für das Ausland
payable à / zahlbar an
edizioni CD - 40121 Bologna
via Boldrini 22 - Italia

ARRETRATI L. 3.000 cadauno
Raccoglitori per annate L. 8.000 (abbonati L. 7.200) + L. 2.000 spese spedizione.

MODALITÀ DI PAGAMENTO: assegni personali o circolari, vaglia postali, a mezzo conto corrente postale 343400.
Per piccoli importi si possono inviare anche francobolli.

STAMPA - FOTOCOMPOSIZIONE FOTOLITO
Tipo-Lito LAME - Bologna
via Zanardi 506 - tel. (051) 376105

XELECTRON

Supplemento a CQ elettronica n° 6

SOMMARIO

giugno 1984

CQ REGALA UN PERSONAL COMPUTER

sinclair
Spectrum

tre li ha già regalati ai signori:

- Ignazio PACCES
  via S. Vittore 38 - 20100 MILANO
- Gianluca MERCURI
  via F.A. Pigafetta 84 - 00154 ROMA
- Aldo MARCHETTO via Turati 14 - 10036 SETTIMO TORINESE (TO)

E TU COSA ASPETTI?

CQ assegnerà a suo giudizio altri tre Personal Computers ZX Spectrum tra tutti i Lettori che, entro il 30 giugno prossimo, risponderanno a queste tre domande:

Personal Computer

1) Perché ancora non ce l'hai?

2) Se tu lo avessi, cosa ci faresti?

3) Lo hai mai usato?

4) Conoscevi CQ?

Tagliando che deve obbligatoriamente accompagnare le vostre risposte. XÉ

I5YJI, Francesco Francescangeli

Come promesso, mi ripresento con qualcosina di hardware.
Questa interfaccia per il G5 non vuole essere assolutamente una antagonista di quelle presentate dall'Ing. Becattini, ma solamente una proposta alternativa, partendo sempre dalla mia idea di base che un appassionato di elettronica non rifiuta mai una nuova soluzione al solito problema ma si diverte a realizzarla e, perché no, a migliorarla.
E quest'ultima è la mia proposta.

INTERFACCIA PER REGISTRATORE, E ALTRI TRASTULLI

Breve storia del marchingegno.

Visto lo schema dell'interfaccia per G5 su **XÉLECTRON**, e notata con angoscia l'omissione dell'elenco componenti, superato un quarto d'ora di shock, mi rimboccai le maniche (e il cervello!) e cominciai a buttar giù lo schema del circuito stampato ("ai componenti penso poi"), assumendo per gli operazionali un LM324.

Fotoinciso, inciso e forato che fu, cominciarono le ipotesi sui componenti, con relative prove "su strada". Non vi dico le ore passate per convincere il 74121 che doveva emettere quei dannati impulsi! Il fatto è, scopersi poi, che con alimentazione singola a +5 V, un operazionale tira fuori un segnale con un'ampiezza di 2,5 V, e questi non sono sufficienti a smuovere il testardo 74121 (almeno i miei tre esemplari).

Confesso che mi prese uno sconforto...

Ripresomi, mi venne in mente di far lavorare l'operazionale con i ±5, come è sua

abitudine, ed effettivamente dei risultati li ottenni, ma con una estrema criticità nella regolazione del volume del registratore. Inoltre non vi dico per trovare i valori di R_8 e C_3 sul 74121! Valori che poi mi lasciavano margini molto stretti di VERIFYnn.

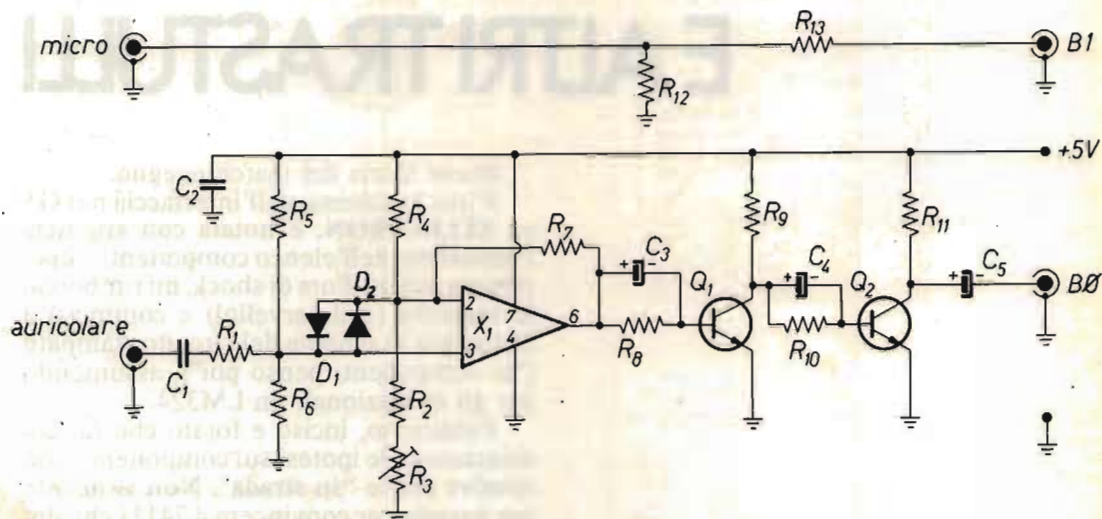
Dato che non potevo sopportare di sottostare alle bizze di una mezza manciata di componenti, misi tutto da parte e cominciai a buttar giù uno schemino di base, talmente classico da inorridire, ma proprio per questo promettente. Fatto lo squadratore d'ingresso, pensai di aggiungere un ancor più classico transistor switch per elevare il segnale ai +5 V appetibili alla Z80PIO: sorpresa! ERR DI LETTURA praticamente fisso sul monitor. E allora? Altro stadio uguale al precedente per reinvertire il segnale (eppure dovevo saperlo che uno switch inverte il segnale!) e nuova prova: VERIFY OK. E finalmente non più problemi di volume: a 10, o al massimo che dir si voglia (almeno con il mio PHILIPS N2234), con il controllo dei toni al massimo verso gli acuti.

Una nota importante: funziona senza problemi dalla velocità VERIFY 5 in su (o giù; insomma numeri alti) senza cambiare alcun componente. E non mi sembra un vantaggio da poco.

Comunque, tenuto ai "lavori forzati" per diverse ore non è uscito l'ERR DI LETTURA neanche una volta, giuro!

Confesso che mi sono sentito orgoglioso della mia creatura.

Io ho installato il tutto dentro il contenitore del G5, con un connettore all'esterno, per evitare una eccessiva lunghezza dei fili verso la PIO, notorie antenne eccezionali per i disturbi, ed anche perché si trovano facilmente i cavi già pronti attestati con i connettori più disparati. Niente proibisce però di metterlo esterno, però con cavi schermati, ovviamente.

E passiamo al circuito, anche se c'è poco da spiegare.

R_1, R_9, R_{11}	3,9 kΩ
R_2	6,8 kΩ
R_3	4,7 kΩ, trimmer
R_4, R_5, R_6	10 kΩ
R_7	5,6 kΩ
R_8, R_{10}	2,2 kΩ
tutte al 5%, 1/4 W	

C_1	0,22 µF, poliestere
C_2	0,1 µF, disco
C_3, C_4, C_5	4,7 µF, 16 V_L
Q_1, Q_2	BC237
X_1	µA741
D_1, D_2	1N4148

Il trimmer R_3 va regolato in modo da avere sul piedino 2 di X_1 una tensione pari alla metà esatta della tensione di alimentazione (non fidatevi del regolatore, misuratela), questo perché il 741 è alimentato a tensione singola; ovviamente fate la misura senza segnale in ingresso. Sul piedino 2 ci dovrebbe già essere metà tensione di alimentazione; se fosse un po' differente, a causa della tolleranza delle resistenze, selezionatevi due resistenze che siano uguali di valore il più possibile.

Tutto qui.

E deve funzionare al primo colpo.

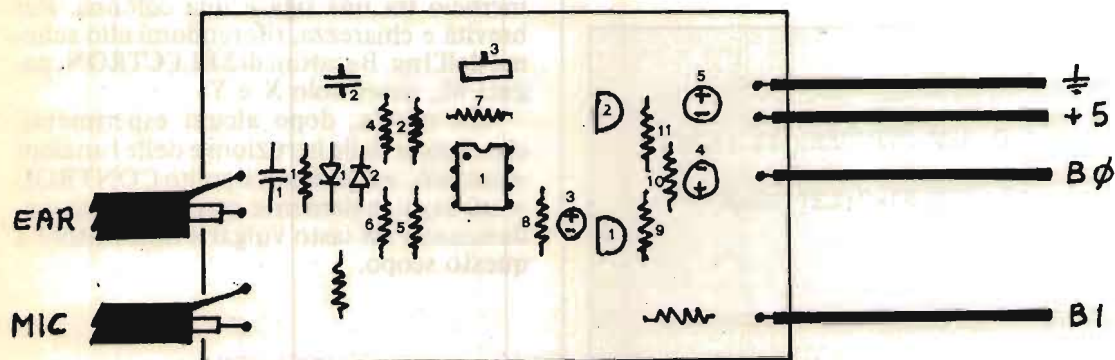
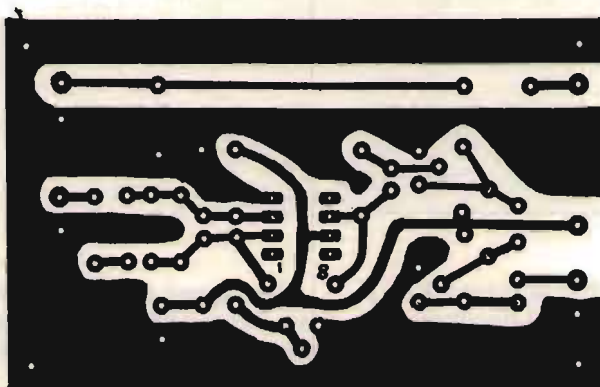
Per la sezione registrazione, restano fermi ovviamente i valori proposti dall'ing. Becattini.

Allego anche per vostra comodità il disegno del circuito stampato e la posizione dei componenti.

Augurandovi buon lavoro, non me ne vado ma, cosparsomi il capo di cenere, debbo fare ammenda di alcuni miei errori nei programmi da me presentati.

Il primo è nel programma per sapere il giorno della settimana.

Purtroppo quando l'ho fatto girare ho scelto delle date che evidentemente non mi hanno dato errori. Ricercandolo su nastró, però, ha cominciato a darmi giorni strani. Dopo alcuni controlli ho potuto stabilire che, a causa di arrotondamenti alla ennesima cifra decimale effettuati dal G5, con alcuni giorni mi usciva fuori un .999999 anziché un intero. Ciò portava il programma ad avere la settimana di 8 giorni, anziché di 7!

Per ovviare a ciò, ho dovuto aggiungere

una riga e modificare alcuni valori in altre. Ecco le righe incriminate, riscritte correttamente:

```
165 IF (D/7-INT(D/7))*7=.999999 THEN LET X=6

170 IF X=2 THEN (uguale alla vecchia)

180 IF X=3 THEN (  "     "     "  )

190 IF X=4 THEN (  "     "     "  )

200 IF X=5 THEN (  "     "     "  )

210 IF X=Ø THEN (  "     "     "  )

220 IF X=6 THEN (  "     "     "  )

230 IF X=1 THEN (  "     "     "  )
```

Ci sono poi due modifiche al programma di conversione hexadecimal-decimale e viceversa del mio ultimo articolo, che io avevo già apportato, ma che, perdonatemi, mi sono dimenticato di apportare anche al listato. Eccole:

```
490 REM OMETTERE IL CLEAR

200 LET N1=Ø:LET N2=Ø
```

Liberatomi così la coscienza di un gran peso, passo a **proporvi una modifica a quelle tastiere basate sulla ROM AY5-2376 o KR2376.**

Come avete visto, la ROM presenta un byte in uscita ogni volta che si chiude un incrocio tra una riga e una colonna. Per brevità e chiarezza, riferendomi allo schema dell'Ing. Becattini di **XÉLECTRON**, pagina 60, citerò solo X e Y.

Ho notato, dopo alcuni esperimenti, che ci sono delle istruzioni e delle funzioni ottenibili, anziché con il solito CONTROL + un tasto, solamente con una manovra, dedicando un tasto vulgaris interruttore a questo scopo.

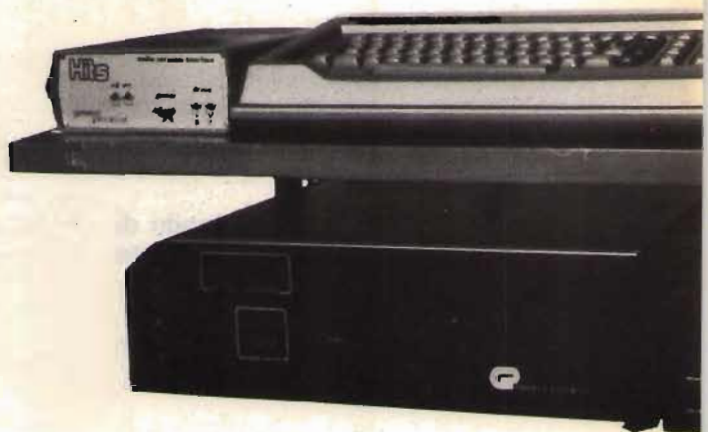
Dato che la modifica, aggiunta di un tasto e due fili da collegare, mi sembra molto semplice e per nulla laboriosa, vi propongo questi nuovi tasti:

INCROCIO		
X	Y	EQUIVALE A
Ø	2	INK
Ø	5	BLANK
Ø	6	GRAPH
Ø	7	NOBLANK
Ø	8	SPDEF
1	Ø	SPFORM
1	5	LOAD
2	4	PAPER
5	9	FORM FEED (cursore inizio quadro e cancella tutto)
6	9	HOR. TAB. (cursore a destra 8 spazi)
6	1Ø	FILE SEPARATOR (cursore inizio quadro)

Un consiglio, anche se ovvio: usate preferibilmente tasti con contatto a reed o comunque un buon tasto da tastiera, non adoperate gli interruttori da pannello, i rimbalzi si sprecano!

Con ciò ho terminato, e, sperando di aver interessato o per lo meno incuriosito qualcuno, vi saluto. Ciao.

FINE

INTER
S
PER

IK4BEM, Daniele Guerzoni

L'interfaccia seriale che viene qui presentata serve a collegare a un microcomputer dotato di interfaccia per stampante parallela una telescrivente seriale consentendo l'utilizzo del software esistente per tutti i comandi di stampa.

FACCIA ERIALE TELESCRIVENTE

Principio di funzionamento

Prima di passare alla descrizione dello schema elettrico, analizziamo brevemente il funzionamento degli standard utilizzati, vale a dire l'interfaccia parallela Centronics e la telescrivente seriale ASCII 110 baud tipo ASR33.

L'interfaccia per stampante parallela Centronics emette i dati in byte formati da 7 bit paralleli e da un segnale di abilitazione per la stampante: il "Data Strobe". La velocità con cui i dati vengono emessi viene condizionata dalla stampante attraverso il segnale "Busy". Questo segnale viene mantenuto alto per il tempo necessario alla stampa.

La telescrivente Teletype ASR33 può operare esclusivamente con dati seriali ASCII con una velocità di 110 baud. Il segnale, che deve essere di tipo start-stop, è composto in 11 bit da un bit di start, 8 bit informativi (di cui uno non utilizzato) e da 2 bit di stop.

Questa macchina, eccettuate le minuscole, può stampare tutto il set dei caratteri ASCII compresi il ritorno carrello e l'interlinea, questi ultimi, però, non in modo automatico. Risulta evidente quindi che se dal calcolatore questi criteri non vengono emessi, eventualità probabile, bisogna tenerne conto nella realizzazione della interfaccia al fine di comandare la stampa in modo corretto.

Altra caratteristica della telescrivente è quella di riconoscere come segnale un loop di corrente di 20 mA.

L'interfaccia seriale qui presentata, oltre alla serializzazione dei dati a 110 baud, provvede, se necessario, a generare il ritorno carrello e l'interlinea automaticamente senza perdere dati significativi in uscita al calcolatore. Può inoltre, dopo aver contato i caratteri emessi verso la telescrivente e aver raggiunto il numero programmato, generare automaticamente il ritorno carrello e l'interlinea evitando così la sovrapposizione di più caratteri a fine linea.

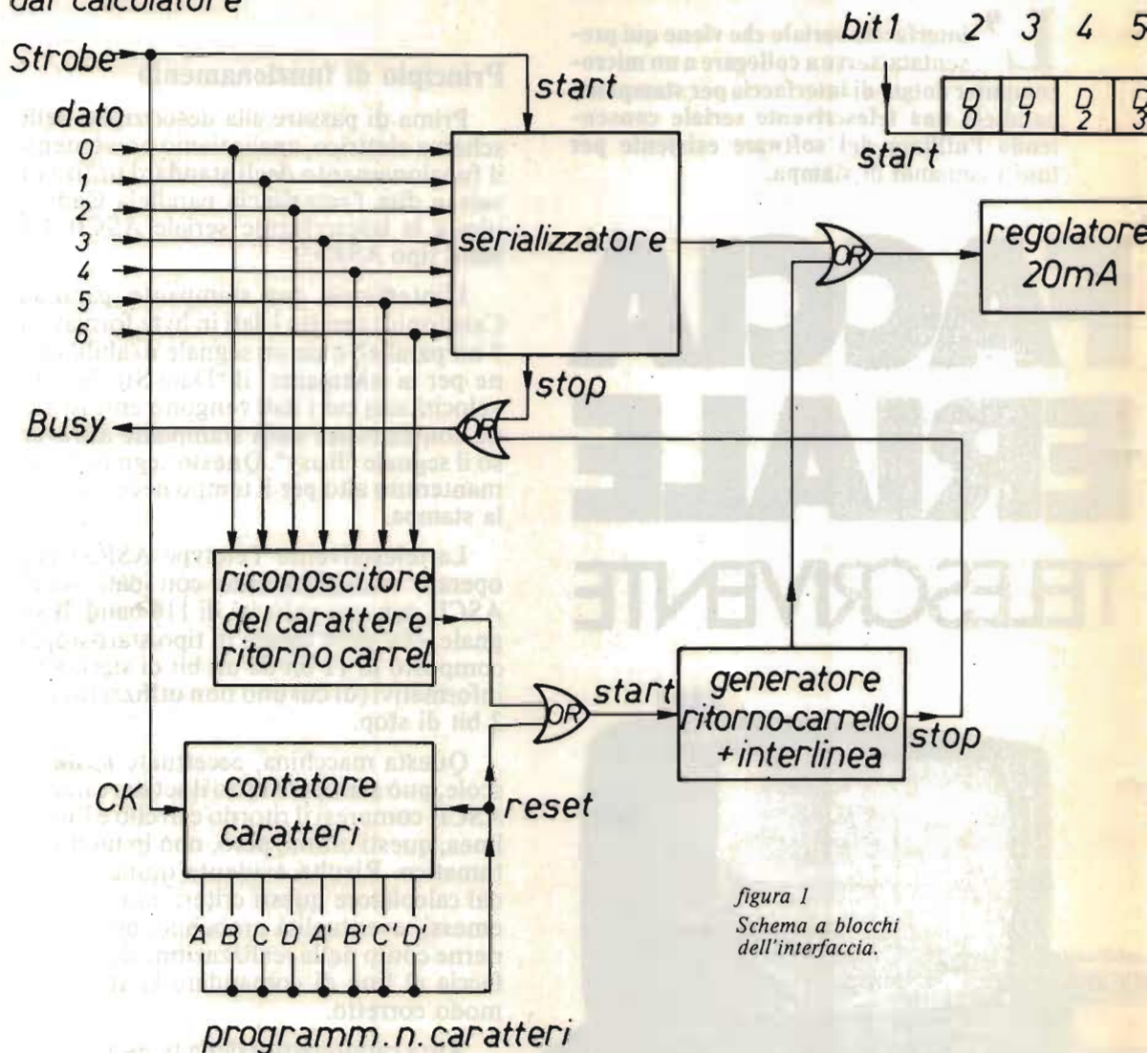



figura 1
Schema a blocchi dell'interfaccia.

L'interfaccia, come si vede dalla figura 1, è composta da un serializzatore, un riconoscitore di ritorno carrello, un generatore di ritorno carrello e interlinea e da un contatore di caratteri.

Schema elettrico

Serializzatore e driver teletype in figura 2.

Questo blocco, il più importante, provvede a multiplexare attraverso l'integrato X_1, i dati in ingresso dato 0-6 in un'unica uscita.

La temporizzazione viene ottenuta tramite l'integrato X_3, un 555 in configurazione oscillatore astabile con la frequenza di 110 Hz, e dal divisore binario X_2 che provvede a comandare il multiplex 4051 con i segnali A B C D.

L'avvio del serializzatore viene comandato dal data strobe disponibile in uscita dal calcolatore e invertito dal nand X_4. Al

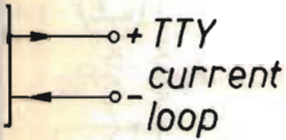
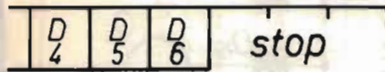



figura 2
Serializzatore e driver TTY.

R_0	4,7 MΩ	$D_1 \div D_{10}$	1N4148
R_1	3,3 kΩ	Q_1	2N1711
R_2	100 kΩ, trimmer	Q_2	BC317
R_3	10 kΩ	Q_3	BC317
R_4	1 MΩ	X_1	4051
R_5	39 kΩ	X_2	4029
R_6	3,9 kΩ	X_3	555
R_7	47 Ω, trimmer	X_4	4011
R_8	4,7 Ω		
R_9	1 kΩ		
R_{10}	330 Ω		
R_{11}	47 Ω		
C_1	5 nF		
C_2	100 nF		
C_3	100 nF		
C_4	47 nF		

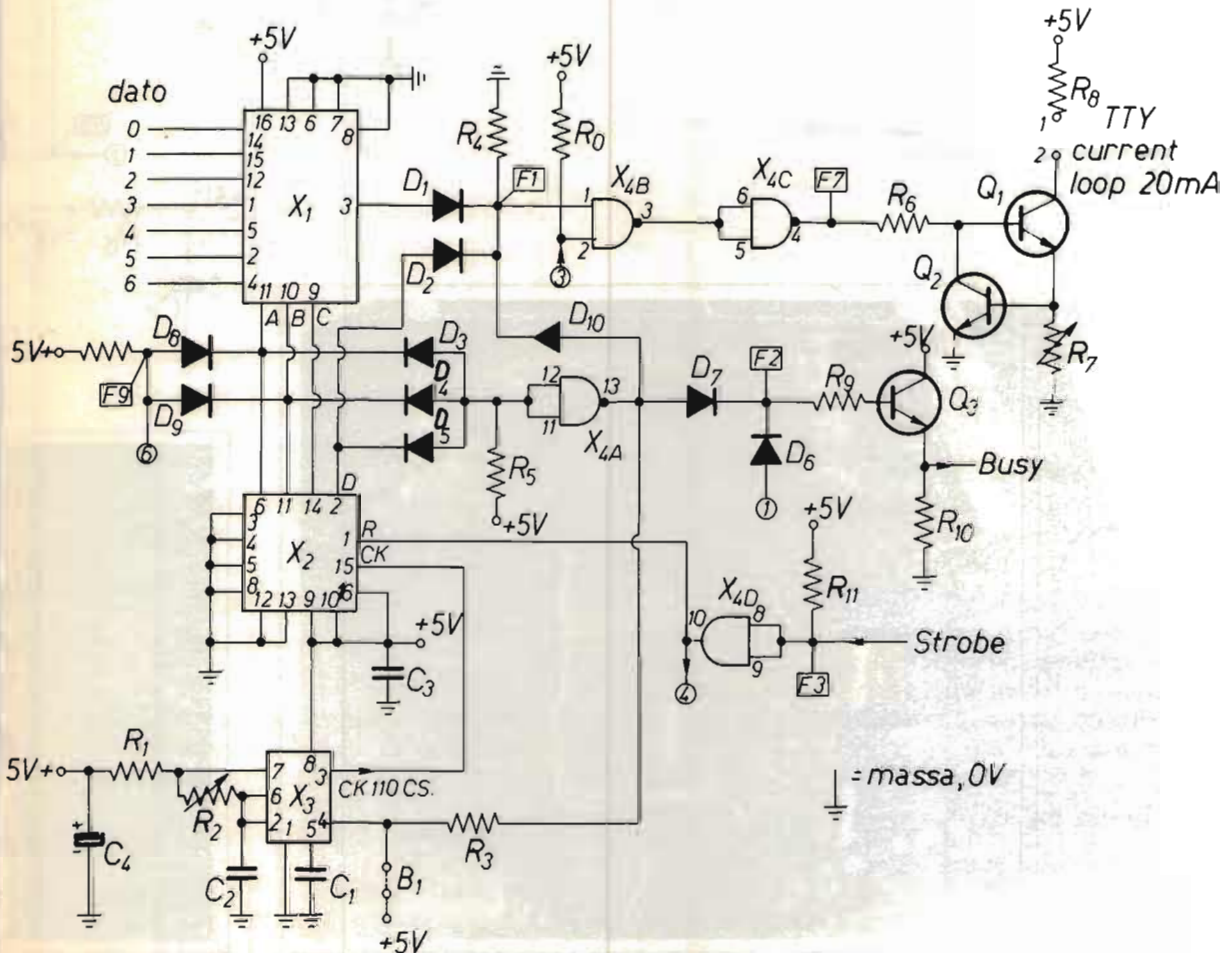
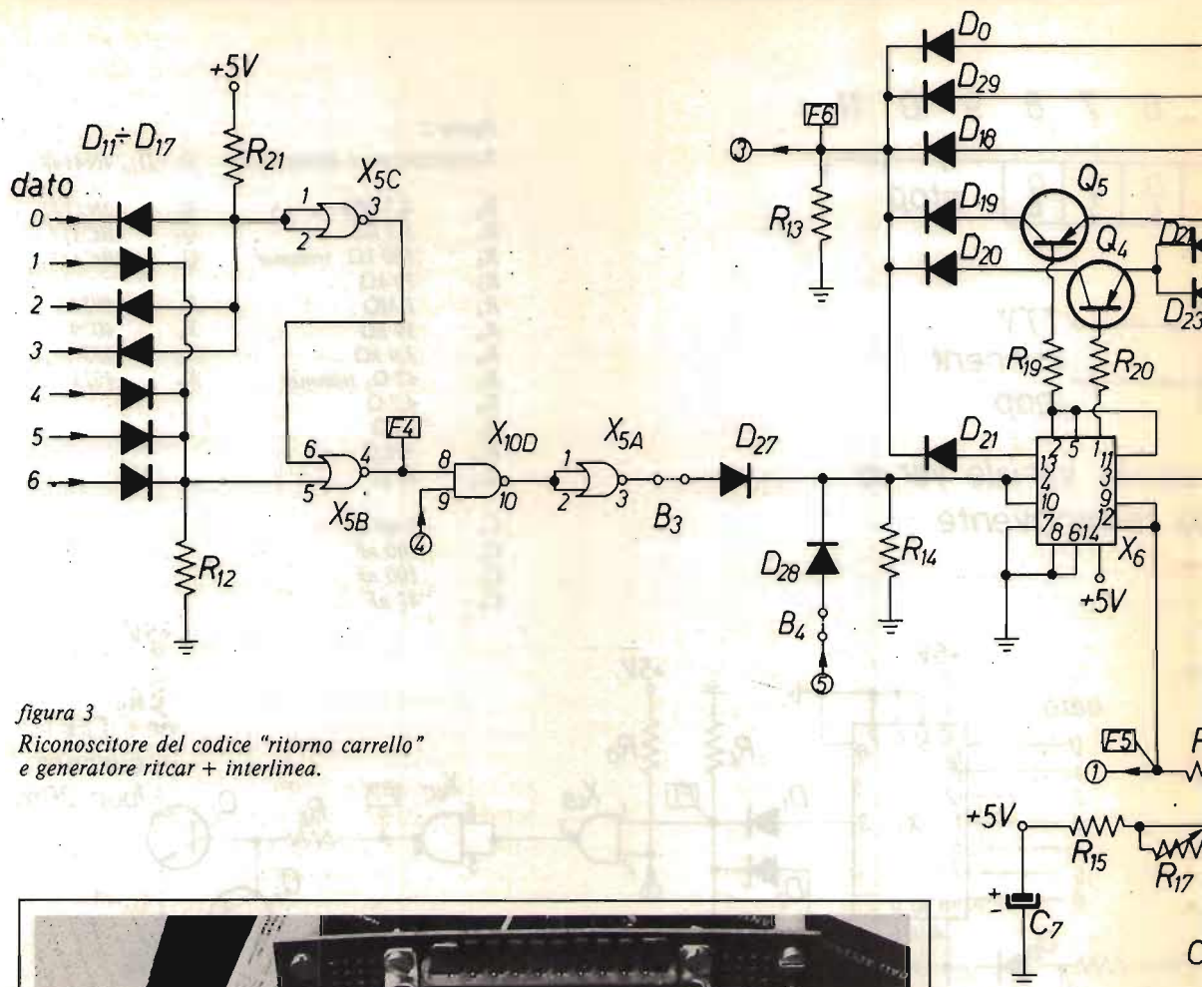
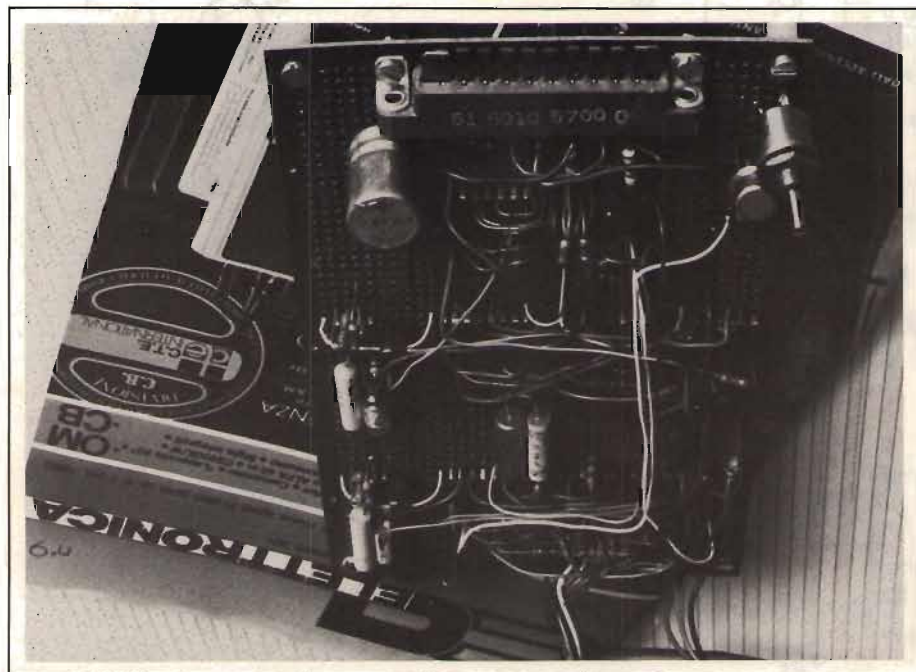
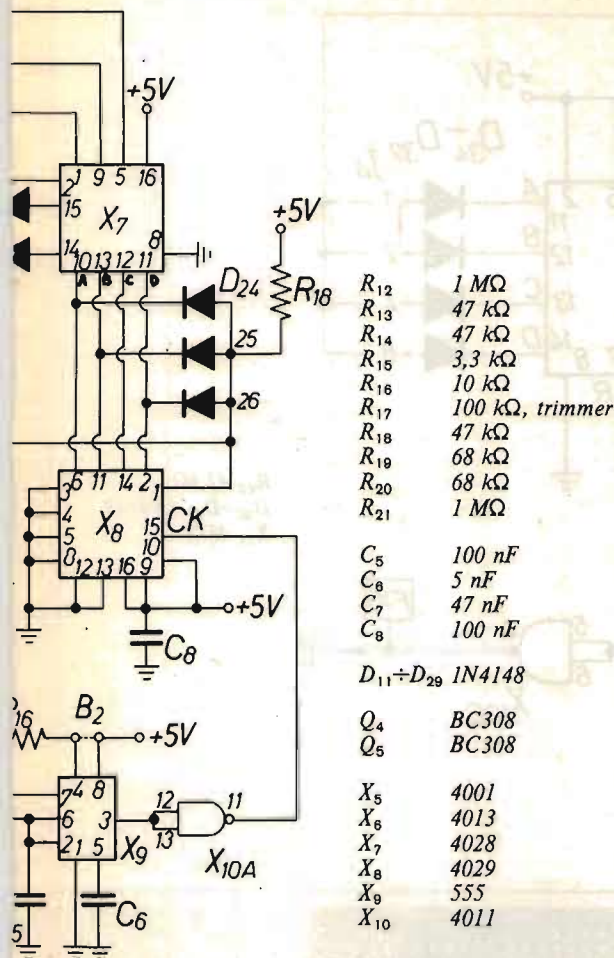



figura 3
Riconoscitore del codice "ritorno carrello" e generatore ritcar + interlinea.

R_{12}	1 MΩ
R_{13}	47 kΩ
R_{14}	47 kΩ
R_{15}	3,3 kΩ
R_{16}	10 kΩ
R_{17}	100 kΩ, trimmer
R_{18}	47 kΩ
R_{19}	68 kΩ
R_{20}	68 kΩ
R_{21}	1 MΩ
C_5	100 nF
C_6	5 nF
C_7	47 nF
C_8	100 nF
$D_{11} \div D_{29}$	1N4148
Q_4	BC308
Q_5	BC308
X_5	4001
X_6	4013
X_7	4028
X_8	4029
X_9	555
X_{10}	4011

termine del ciclo viene generato verso il calcolatore il segnale busy che allo stato basso abilita l'avvio di un nuovo ciclo.

Di questo blocco fa parte anche il generatore a corrente costante che, pilotato dai livelli logici in uscita da X_{4C}, ha il compito di comandare il loop di corrente a 20 mA della teletype.

I transistori Q_1, Q_2 possono, se necessario, essere sostituiti da un optoisolatore che in questo prototipo non si è ritenuto necessario utilizzare. Questo blocco è sufficiente per l'interfacciamento della teletype al calcolatore mentre i blocchi seguenti possono essere omessi se non ne sono richieste le prestazioni.

Il primo blocco opzionale in figura 3, connesso allo schema di figura 2 attraverso i punti indicati nei cerchi numerati, consente tramite l'and formato dai diodi $D_{11} \div D_{17}$ e i nor X_{5B} e X_{5C}, di riconoscere il criterio di ritorno carrello emesso dal calcolatore.

Consente inoltre di avviare il generatore di ritorno carrello e interlinea e di bloccare i dati emessi dal calcolatore tenendo alto il livello del segnale busy per tutto il tempo richiesto dall'esecuzione di queste operazioni. La temporizzazione di questo blocco è ottenuta tramite X_9, un altro 555 in configurazione astabile con frequenza 110 Hz, dal divisore binario X_8 e da X_6, doppio flip-flop D. Per completare il funzionamento del generatore di ritorno carrello e interlinea si fa uso dell'integrato X_7, un decoder BCD/decimale.

La seconda opzione è costituita dal contatore di caratteri programmabile mostrato in figura 4 che, connesso ai precedenti blocchi, consente di attivare il generatore di ritorno carrello e interlinea dopo un numero prefissato di caratteri.

Il blocco è costituito dal doppio contatore decimale X_{11} e dalle sezioni B-C di X_{10} che concorrono alla formazione dei segnali di controllo necessari. Il riconoscimento del numero programmato è ottenuto attraverso l'and formato dai diodi $D_{30} \div D_{37}$.

Per facilitare la comprensione del funzionamento dell'interfaccia, si possono notare in figura 2 le forme d'onda rilevate sul prototipo durante il funzionamento. Più precisamente, sempre in figura 2, al tempo 2, F_1 mostra un ritorno carrello, in

R_{22} 47 kΩ
$D_{30} \div D_{37}$ 1N4148
X_{11} 4518

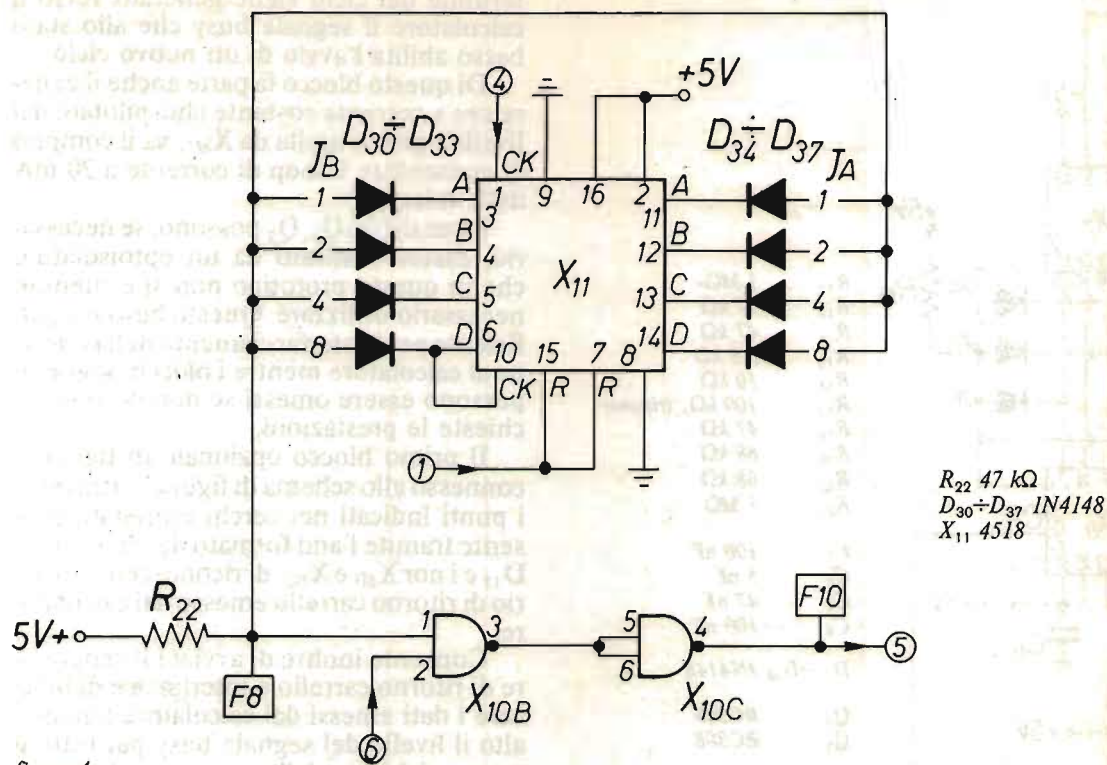



figura 4
Contatore di caratteri programmabile.

tempi: 1 2 3 4 5 6 7 8

F_1 F_2 F_3 F_4 F_5 F_6 F_7 F_8 F_9 F_{10}

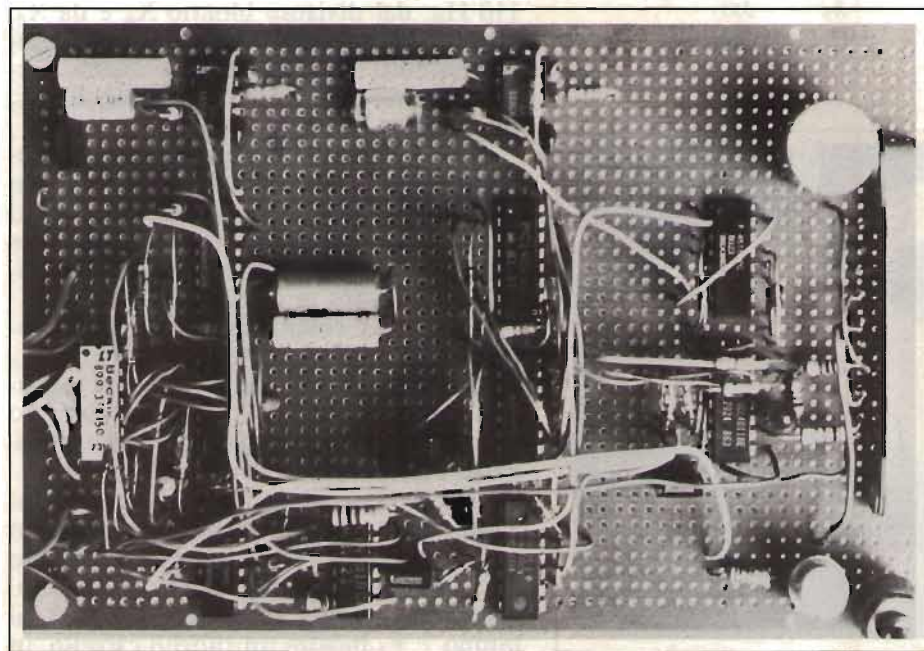
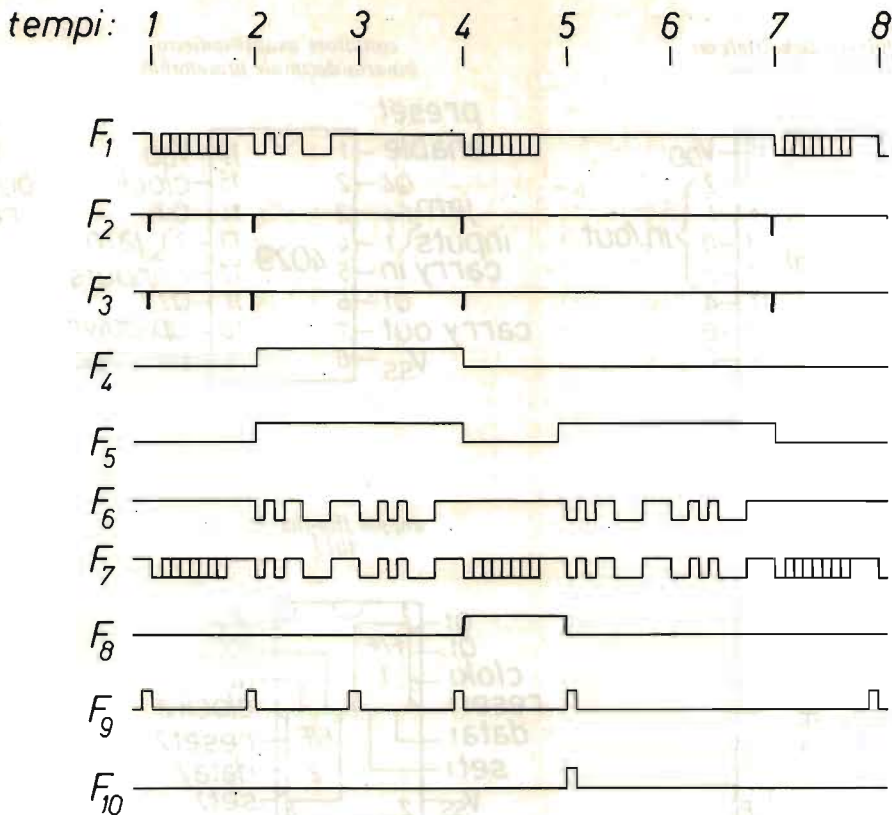



figura 5

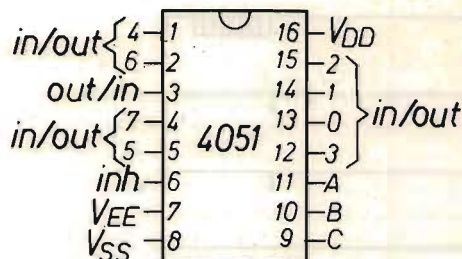
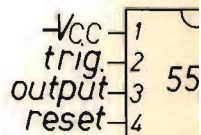
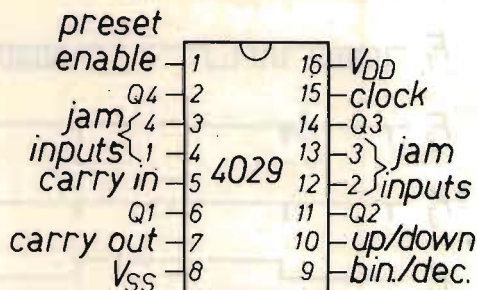
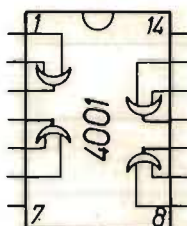
ASCII codice 13.

In F_4 il riconoscimento di questo criterio e in F_6 la generazione del ritorno carrello e interlinea. Osservando ancora la figura 2, F_8 ci indica il raggiungimento del numero dei caratteri programmati da parte del contatore. F_8, insieme al segnale F_9 che indica il completamento dell'ultimo carattere da stampare forma il segnale F_{10} che, dal tempo 5 al tempo 7, comanda un ciclo di ritorno carrello e interlinea.

Tarature e predisposizioni

Le sole tarature richieste sono quelle per la messa in frequenza dei due oscillatori X_3, X_9 e la regolazione della corrente del driver della teletype.

Per la messa a punto degli oscillatori si deve misurare una frequenza di 110 Hz sul pin 3 degli integrati X_3, X_9 dopo aver eseguito i ponticelli B_1 - B_2 che ne forzano il

multiplexer/demultiplexer singolo a 8 canali

contatore avanti/indietro binario/decimale presettabile

quadruplo nor a 2 ingressi

doppio flip-flop D 4013

decodi BCD - d

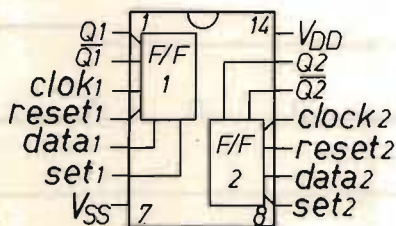
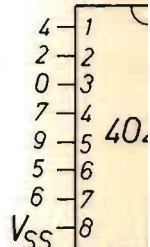
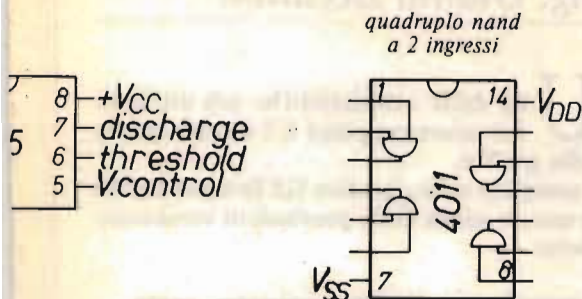
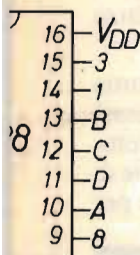
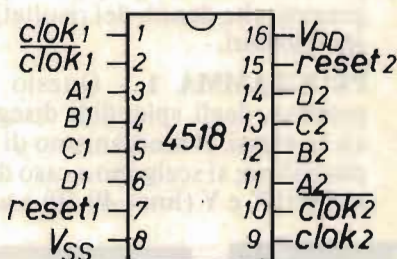
quadruplo nand a 2 ingressi

...catore ...ecimale

16 VDD
15 3
14 1
13 B
12 C
11 D
10 A
9 8

doppio contatore decimale

clok1 1 — 16 VDD
clok1 2 — 15 reset2
A1 3 — 14 D2
B1 4 — 13 C2
C1 5 — 12 B2
D1 6 — 11 A2
reset1 7 — 10 clok2
Vss 8 — 9 clok2

4518

funzionamento agendo su R_2 e R_{17}.

Il driver della teletype deve erogare 20 mA e la regolazione viene effettuata, col tester connesso ai punti 1 e 2, agendo su R_7.

Predisposizioni

B_1, B_2, se eseguiti, forzano al funzionamento gli oscillatori X_3 e X_9; vanno rimossi dopo la taratura.

B_3, se eseguito a ogni ritorno carrello generato dal calcolatore, fa si che venga aggiunto dall'interfaccia il criterio interlinea.

B_4, se eseguito, comanda il posizionamento della teletype su una nuova linea dopo il raggiungimento del numero programmato.

J_A i ponticelli 3-8 programmano in codice binario le decine dei caratteri da stampare.

J_B i ponticelli 1-8 programmano le unità dei caratteri da stampare.

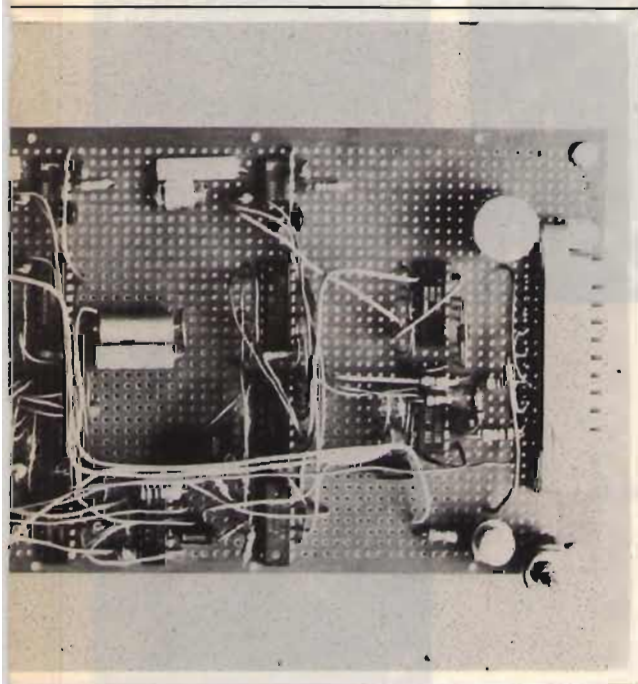
Conclusione

L'interfaccia è stata realizzata per adattare una teletype ASR33 a una uscita per stampante parallela di un Child Z. È stata collegata a quest'ultimo direttamente ai port input/output e ne utilizza la alimentazione a +5 V.

Il montaggio del prototipo dell'interfaccia è stato effettuato su di una basetta forata eseguendo con fili la connessione fra i componenti.

Il funzionamento, dopo diversi mesi di utilizzo, non ha presentato alcun inconveniente.

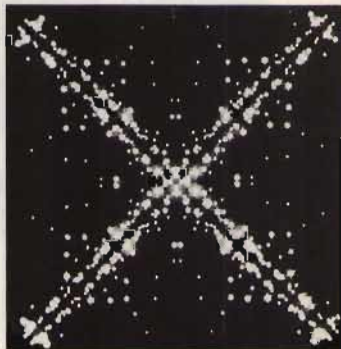
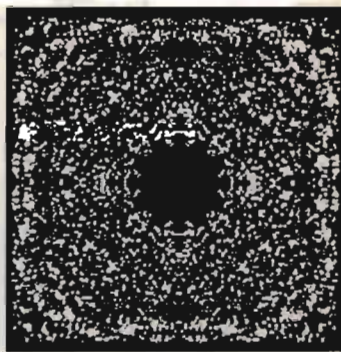
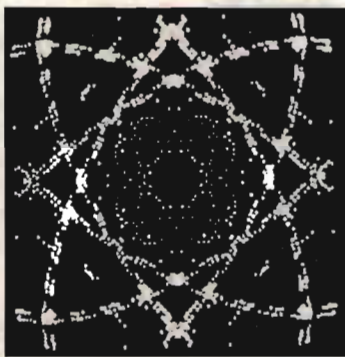
Augurandomi di essere stato di qualche utilità a qualche altro lettore, porgo cordiali saluti. Mi ritengo a disposizione per qualsiasi chiarimento.

FINE

ing. Gianni Becattini

Una delle caratteristiche più attraenti del microcomputer G5 è certo quella della grafica.
Il semplice ed economico G5 fornisce infatti in questo senso delle prestazioni veramente buone.

G5 ARTISTA

Per dare un esempio di ciò, ecco che vi presento alcuni semplici esempi di programmi che danno dei risultati abbastanza spettacolari.

PROGRAMMA 1 - Questo programma produce degli splendidi disegni "riflessi" su se stessi. Il meccanismo di base è semplicissimo; si scelgono a caso due coppie di numeri X e Y (linee 40-50) e un colore per

Disegni ottenuti con il programma 1.

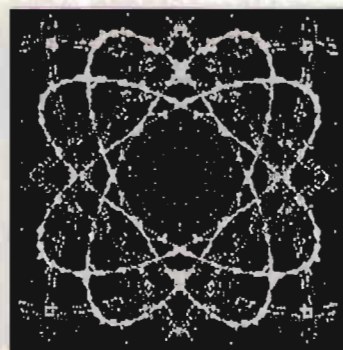
il tracciamento. Quindi si valutano due funzioni f_x (x,y) e f_y (x,y) (linee 60-70) che rappresentano le coordinate del punto da illuminare. Successivamente si usano f_x e f_y come x e y per valutare le nuove f_x e f_y e così via ciclicamente. Dopo un certo numero di passi, anch'esso casuale (linea 55) si scelgono una nuova coppia di partenza X, Y e un nuovo colore per la INK. L'effetto di riflessione è ottenuto accendendo, per ogni coppia elaborata, una serie di punti simmetrici (frasi da 95 a 124). Variando le funzioni f_x e f_y (linee 60-70) si ottengono i diversi effetti; in ogni caso, data l'abbondanza di funzioni RND, ad ogni esecuzione l'effetto può differire sensibilmente anche a parità di funzioni.

Per mia sbadataggine ho dimenticato di segnare le funzioni scelte via via che facevo le foto: ecco comunque un elenco da usare per base di esperimenti, da sostituire a quelle presentate nella lista del programma:

```
X=X-Y/2        X=X-Y/8
Y=Y+X/2.1      Y=Y+X/8

X=X-Y/3        X=X+Y/2
Y=Y+X/1.5      Y=Y-X/2

X=X-2*Y        X=X-Y/8
Y=Y+X/1.1      Y=Y-X/8

X=X-Y/2        X=X-(Y/1.1)
Y=Y+Y/4        Y=Y-X/2

X=X-Y
Y=Y+X/2

X=X-2*Y
Y=Y+X/3

X=X-2Y
Y=Y+2*X
```

```
1 LET G=2
10 GRAPH
11 GRAPH
15 LET A=255
20 ORG 127,95
40 LET X=RND(0)*A
50 LET Y=RND(0)*A
52 INK RND(1)*15
55 FOR I=1 TO 70*RND(0)
60 LET X=X-Y/8
70 LET Y=Y+X/8
75 IF ABS(Y)>191 THEN 130
76 IF ABS(X)>191 THEN 130
80 LET L=ABS(X)/G
90 LET M=ABS(Y)/G
95 SET L,M
100 SET L,-M
110 SET -L,-M
120 SET -L,M
121 SET M,L
122 SET -M,L
123 SET -M,-L
124 SET M,-L
130 NEXT
140 GOTO 40
```

Programma 1

Altre funzioni sono lasciate alla vostra fantasia.

PROGRAMMA 2 - Questo programma produce invece degli stupendi disegni che sembrano l'immagine generata da una retta in movimento; complesse ma a loro modo regolari superfici geometriche di grande effetto. Apparentemente il crearle è complicatissimo; basta conoscere il trucco per rendersi conto del contrario.

Si tratta semplicemente di questo:

1) creare una palla che rimbalza contro i lati dello schermo (vedere l'esempio sul mio libro "IL COMPUTER È FACILE" edito dalle edizioni CD a pagina 10.12);

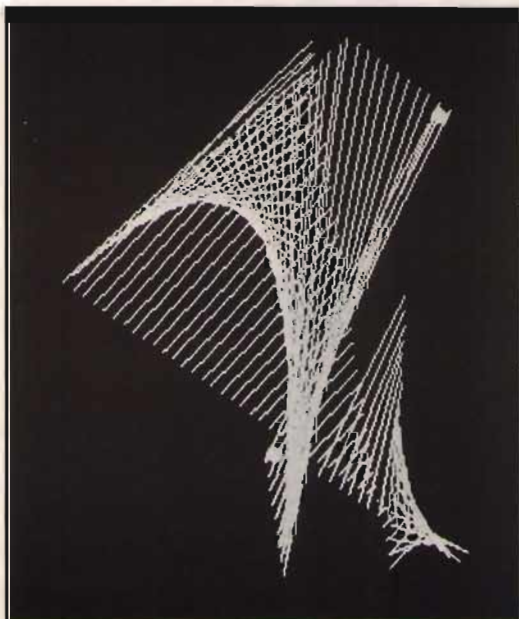
2) crearne un'altra identica alla prima che rimbalza per conto proprio; si hanno così due palle indipendenti che vagano sullo schermo con legge abbastanza casuale;

3) unire, a certi intervalli di tempo, le due palle con un segmento (linea 120). La successione dei segmenti rappresenta proprio il disegno che desideravamo produrre.

Anche in questo caso vale una notevole casualità; può quindi avvenire che un disegno venga bello e uno brutto, così come può avvenire che un disegno inizialmente bello si deteriori col procedere del tracciamento o viceversa.

Programma 2

```
5 GRAPH
6 GOSUB 2000
7 INK 11
10 LET X=RND(0)*255
20 LET Y=RND(0)*191
30 LET L=RND(0)*255
40 LET M=RND(0)*191
50 LET U=15:LET V=7
100 GOSUB 1000
115 LET N1=N1-1
116 IF N1=0 THEN GOSUB 1000
119 SPMOV 0,X,Y:SPMOV 1,L,M
120 PLOT X,Y,L,M
140 IF X+A>255 THEN LET A=-A
145 IF X+A<0 THEN LET A=-A
150 IF Y+B>191 THEN LET B=-B
155 IF Y+B<0 THEN LET B=-B
160 IF L+C>255 THEN LET C=-C
165 IF L+C<0 THEN LET C=-C
170 IF M+D>191 THEN LET D=-D
175 IF M+D<0 THEN LET D=-D
180 LET X=X+A:LET Y=Y+B
```

Figure ottenute con il programma 2.

```
90 LET L=L+C:LET M=M+D
10 LET C0=RND(0)*200
30 IF C0=1 THEN GOTO 10
35 GOTO 115
000 LET A=RND(0)*U-V
010 LET B=RND(0)*U-V
020 LET C=RND(0)*U-V
030 LET D=RND(0)*U-V
040 LET N1=RND(0)*20+10
050 RETURN
000 SPCOL 0,15
010 SPCOL 1,15
020 SPTYP 0,0,2
030 SPDEF 0,0,OF
040 SPDEF 0,1,OOOXXF
050 SPDEF 0,2,OOXXXXF
060 SPDEF 0,3,OXXXXXXF
070 SPDEF 0,4,OXXXXXXF
080 SPDEF 0,5,OOXXXXF
090 SPDEF 0,6,OOOXXF
120 SPFORM 1,0
130 RETURN
```

Programma 3

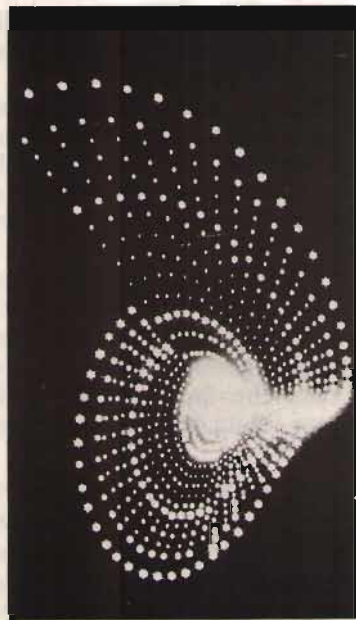
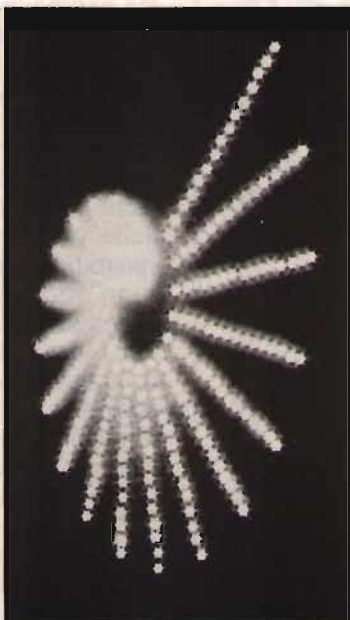
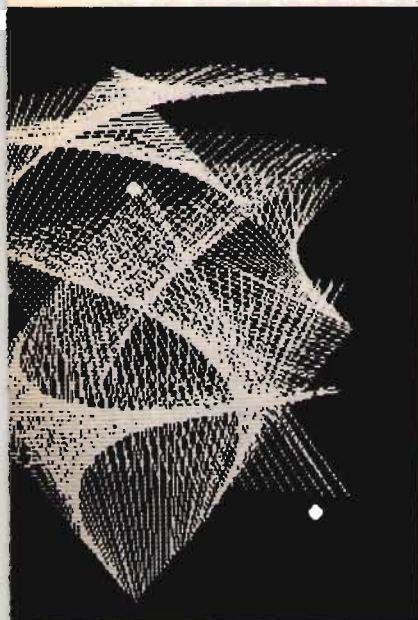
```
1 GRAPH
10 FOR A=5 TO 19
20 FOR J=1 TO 7 STEP .3
30 SET A*J*COS(J)/2+110,A*J*SIN(J)+100
40 NEXT
50 NEXT
```

Programma 4

```
10 GRAPH
50 FOR A=2 TO 19
60 FOR J=1 TO 7 STEP .1
70 SET A*(J++4*SIN(J))*COS(J)+110,A*(J+2
*SIN(J))*SIN(J)+100
80 NEXT
85 INK 15*RND(0)
90 NEXT
```

PROGRAMMI 3 e 4 - Questi non sono farina del mio sacco e li riporto solo a titolo di esempio. Il programma 4 è molto lento. L'effetto non è sconvolgente per nessuno dei due ma pur sempre interessante.

Figura ottenuta con il programma 3.

Figura ottenuta con il programma 4.

Anche su carta la grafica del G5... e a 4 colori

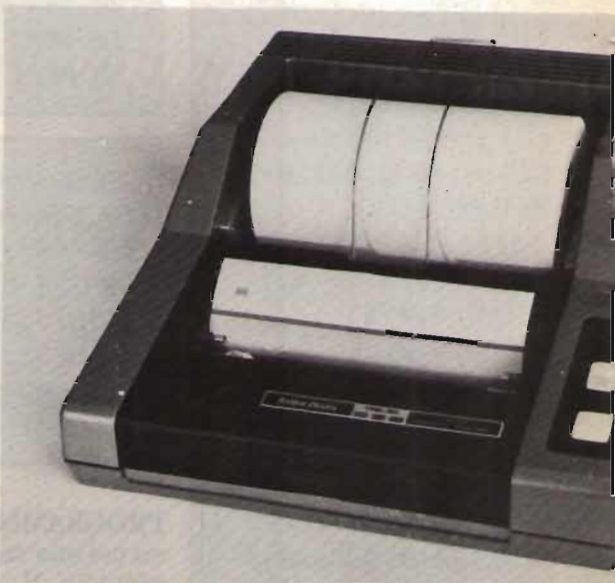
Tutti coloro che hanno avuto modo di adoperare il G5 hanno potuto apprezzare le sue elevate caratteristiche grafiche. Da quando è uscito su **XÉLECTRON** l'articolo sulla interfaccia per stampante, molti mi hanno chiesto suggerimenti sul modello a mio avviso più indicato.

Oggi esistono in commercio moltissimi modelli diversi di stampante; alcune di esse vengono definite "grafiche" anche se sono piuttosto difficili da usare stante la necessità di accedere a singoli punti di una matrice e senza possibilità di far arretrare la carta. Stanno tuttavia diffondendosi sempre più le stampanti "a penna" dopo la loro introduzione che, perlomeno sulle calcolatrici evolute, mi sembra che sia avvenuta ad opera della giapponese Sharp. In esse la scrittura avviene mediante una speciale penna a sfera che, grazie al movimento del carrello e alla possibilità del rullo di girare anche all'indietro, scrive con tratto perfettamente "continuo" come le normali penne a sfera. In virtù delle ridottissime dimensioni dell'organo di scrittura, queste stampanti sono di solito munite di ben quattro "refill", montati su un tamburo girevole tipo "Colt" comandabile da programma. Si ottiene così la scrittura in quattro diversi colori e con una qualità davvero buona.

Questa architettura si presta ottimamente a conseguire simultaneamente due obiettivi; quello della normale stampante e quello del plotter, di quella meravigliosa macchina cioè che produce perfetti disegni su carta.

Chi non ha mai, più o meno segretamente sognato di possedere un plotter? Forse solo coloro che non l'hanno mai visto in funzione con la sua penna veloce e sicura che quasi sembra guidata da una mano fatata (mi sono lasciato trasportare, eh?).

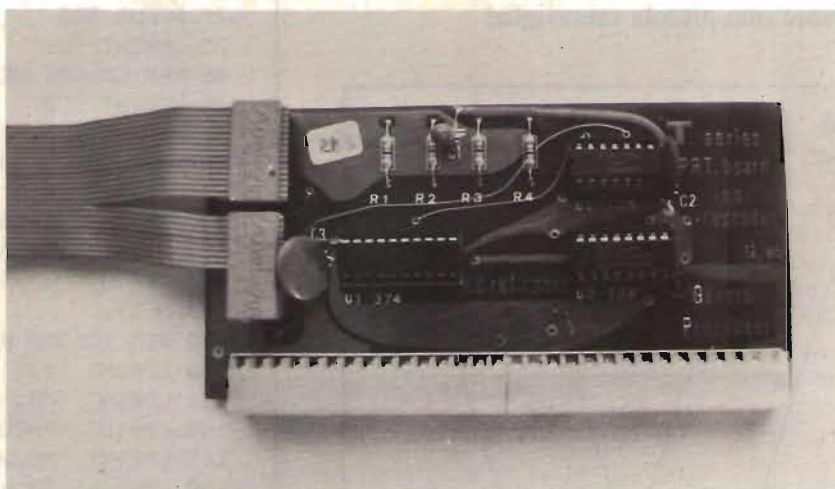
Queste nuove stampanti "Colt", come le chiamo io in onore al vecchio West e al loro tamburo girevole, hanno inoltre una stupefacente risoluzione: il passo minimo di cui può essere mossa la penna è di soli due decimi di millimetro. Anni fa, un plotter con questa risoluzione costava fior di milioni (di allora).

Sul mercato esistono vari modelli di "Colt".

La stampante di cui si parla nell'articolo.

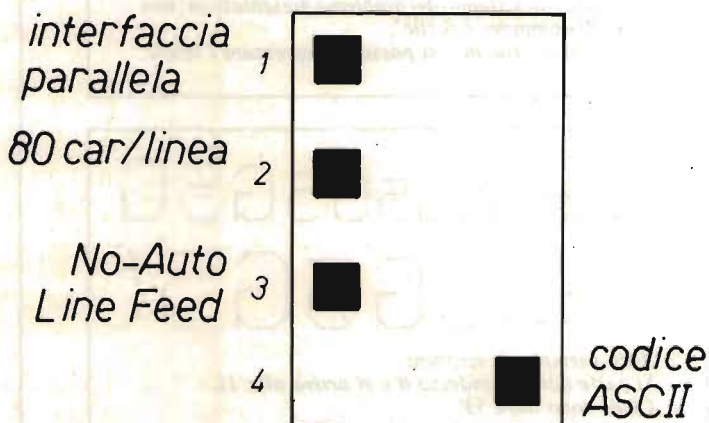
La mia scelta è caduta sulla **CGP-115** della **Radio Shack,** non per ragionamento ma perché, per una serie di motivi, ne ero entrato in possesso. Malgrado la mia tiepida simpatia verso la Casa americana, sono rimasto davvero entusiasta delle sue prestazioni, grazie anche al mio microprocessore incorporato che la rende alquanto "intelligente" e facilissima da usare. La carta viene vista come un piano ove si trovano delle coordinate cartesiane con l'origine spostabile a piacere. Le direzioni degli assi sono quelle convenzionali, per cui non è necessario fare calcoli o contorsioni mentali. I movimenti sono comandabili in senso assoluto (cioè rispetto all'origine degli assi) o in senso relativo alla posizione corrente della penna. Per tracciare una retta basta il seguente comando

LPRINT "D200,–20"

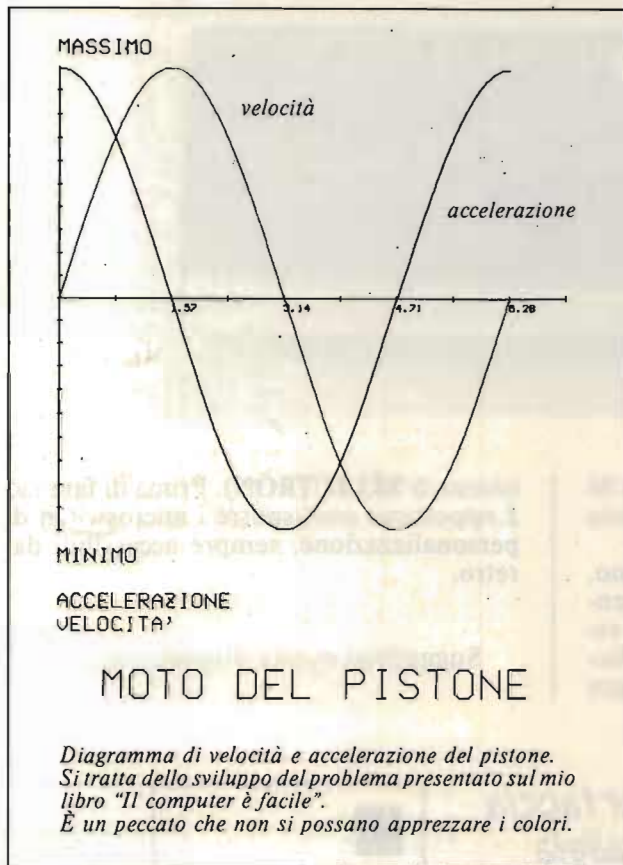
perché la penna si sposti, tracciando, dalla posizione attuale al punto di coordinate X=200, Y=–20. In direzione orizzontale

L'interfaccia stampante del G5.

sono disponibili 480 punti di scrittura su 96 mm. In direzione dell'asse Y non ci sono limiti.

L'interfacciamento al G5 è facilissimo. Basta innestare il connettore tipo Centronics nell'apposito alloggiamento sul retro della CGP-115 (è necessaria l'interfaccia stampante già presentata sul numero scorso di **XÉLECTRON**). Prima di fare ciò è opportuno predisporre i microswitch di personalizzazione, sempre accessibili dal retro.

Suggerisco questa disposizione:

Riporto qualche figura con degli esempi di elaborati.

Sono rimasto davvero entusiasta di questa stampante "Colt". È prevedibile che altri Costruttori entrino presto in lizza su questo tipo di oggetti, per cui prima dell'acquisto documentatevi bene per avere un panorama aggiornato. In ogni caso il G5 dotato di una stampante a quattro colori è veramente una piccola meraviglia!

Diagramma di velocità e accelerazione del pistone. Si tratta dello sviluppo del problema presentato sul mio libro "Il computer è facile". È un peccato che non si possano apprezzare i colori.

G5G5G5G5G5G5G5G
G5G5G5G5G5(

Altro esempio di scrittura. Si parte dalla grandezza 0 e si arriva alla 11. Ce ne sono altre 53!

```
10 LPRINT CHR$(18)
15 LPRINT "C0"
30 LPRINT "I"
40 LPRINT "X1,50,9"
50 LPRINT "H"
60 LPRINT "M0,200"
70 LPRINT "X0,-20,20"
71 GOSUB 500
75 LPRINT "S1"
85 REM COLORE VERDE
86 LPRINT "C2"
87 LPRINT "M0,-290"
88 LPRINT "PVELOCITA'"
89 LPRINT "M0,0"
90 FOR I=1 TO 400
100 LET Y=SIN(I*3.14/200)*200
110 LPRINT "D",STR$(I),",",STR$
120 NEXT
130 REM COLORE ROSSO
140 LPRINT "C3"
141 LPRINT "M0,-270"
143 LPRINT "PACCELERAZIONE"
145 LPRINT "M0,200"
150 FOR I=1 TO 400
160 LET Y=COS(I*3.14/200)*200
170 LPRINT "D",STR$(I),",",STR$
180 NEXT
185 LPRINT "Q0"
190 LPRINT "M0,210"
200 LPRINT "S1"
210 LPRINT "C0"
220 LPRINT "PMASSIMO"
230 LPRINT "M0,-230"
240 LPRINT "PMINIMO"
250 LPRINT "M40,-350"
260 LPRINT "S3"
270 LPRINT "C1"
280 LPRINT "PMOTO DEL PISTONE"
400 STOP
500 LPRINT "S0"
510 LPRINT "M100,-12"
520 LPRINT "P1.57"
530 LPRINT "M200,-12"
540 LPRINT "P3.14"
550 LPRINT "M300,-12"
560 LPRINT "P4.71"
570 LPRINT "M400,-12"
580 LPRINT "P6.28"
590 RETURN
1000 LPRINT "H"
```

Lis pre fun con È u zio Il l rat

a del programma che ha generato il diagramma :edente. L'ho fatto in fretta, per la smania di veder :ionare la stampante, per cui non è certo da imitare :e esempio di brillante programmazione. :ile tuttavia per chi vuole decifrare la logica di fun- ıamento del microcomputer interno alla CGP-115. st è stato eseguito con due delle 64 grandezze di ca- :re selezionabile.

```
INT(Y))

INT(Y))

145 LPRINT "M0,200"
150 FOR I=1 TO 400
160 LET Y=COS(I*3.14/200)*200
170 LPRINT "D",STR$(I),",",STR$(INT(Y))
180 NEXT
185 LPRINT "Q0"
190 LPRINT "M0,210"
200 LPRINT "S1"
210 LPRINT "C0"
220 LPRINT "PMASSIMO"
230 LPRINT "M0,-230"
240 LPRINT "PMINIMO"
250 LPRINT "M40,-350"
260 LPRINT "S3"
270 LPRINT "C1"
280 LPRINT "PMOTO OEL PISTONE"
400 STOP
500 LPRINT "S0"
510 LPRINT "M100,-12"
520 LPRINT "P1.57"
530 LPRINT "M200,-12"
540 LPRINT "P3.14"
550 LPRINT "M300,-12"
560 LPRINT "P4.71"
570 LPRINT "M400,-12"
580 LPRINT "P6.28"
590 RETURN
1000 LPRINT "H"
```

Lo stesso diagramma del primo grafico ma con una gobba del SIN(X) parzialmente annerita (verde nell'originale). Si noti la precisione e la regolarità del tratto.
Ogni eventuale imperfezione è dovuta solo alla riproduzione e alla stampa.

FINE

ing. Gianni Becattini

G5 SPERIMENTARE

Il microcomputer G5, a suo tempo presentato sulle pagine di CQ (vedi n. 1 e 2/83) è gia stato duplicato in circa 250 esemplari con buona soddisfazione dei rispettivi autocostruttori.
Per quanto mi concerne, non un solo esemplare di G5 è al momento in panne.

Già quindi cominciano a pervenire risultati di esperienze svariate.

Vediamo oggi quelle del sig. **Claudio Redolfi** che ha costruito ben tre esemplari di G5.

Lascio quindi a lui la parola:

Ho realizzato con soddisfazione due esemplari di G5 per dei miei amici e uno per me. Dopo prolungata sperimentazione ho messo a punto alcune migliorie che illustro e propongo:

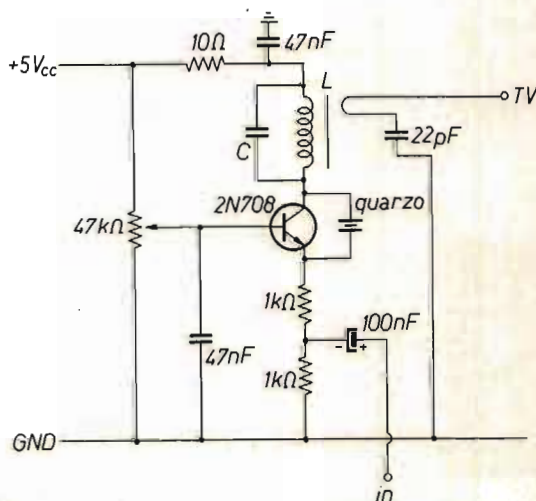
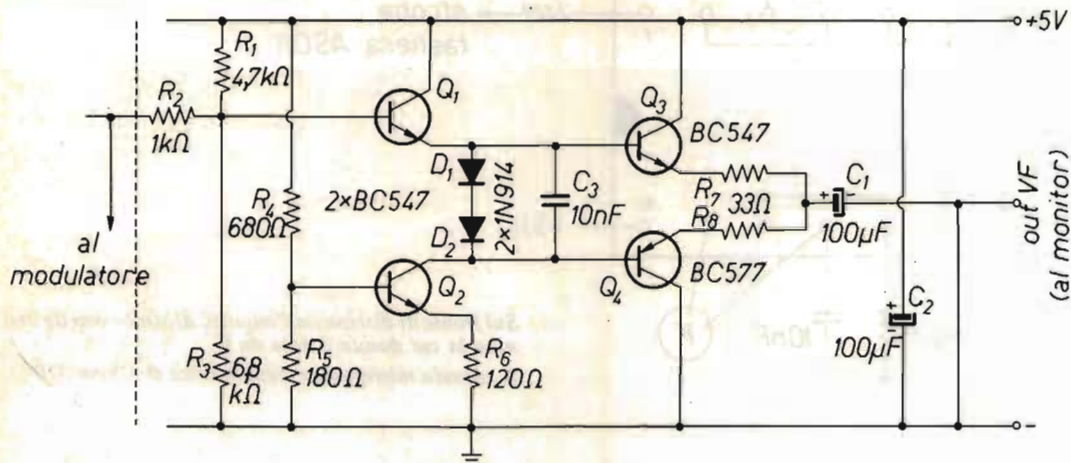
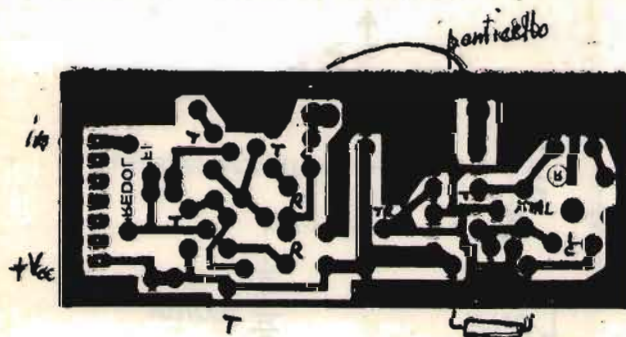



figura 1

Modulatore video (segnale in banda III^a^).

quarzo 27,125 MHz c.a.
C 5,6 pF
L 6 spire spaziate su Ø 6 mm, filo Ø 0,3 mm con nucleo e link 1 spira lato freddo
Al posto del 2N708 funziona meglio un BFR34.
N.B. - Togliere il condensatore da 100 nF in caso il modulatore venga usato con il separatore di pagina 29.

MODULATORE VIDEO *(figura 1) - Poiché non tutti possiedono un monitor, ho realizzato un semplice modulatore bianco e nero per vedere l'uscita del G5 su un qualunque televisore. Lo stadio separatore è opzionale e l'ho previsto per assicurare maggiore protezione al costosissimo TMS9929. Sto lavorando a un modulatore a colori che illustrerò ai lettori di* **CQ** *non appena pronto.*

Separatore

Q_1 e Q_2 anche BC171 o BC107
Q_3 e Q_4 anche BC107/BC177

Modulatore video BN + stadio separatore, lato rame.

Lato componenti.

BIP PER TASTIERA *(figura 2) - Desideravo fare in modo che alla pressione di ogni tasto si udisse un "bip". Ho risolto il problema con il semplice circuito che riporto* (secondo me è una cosa ossessiva... nota del Becattini).

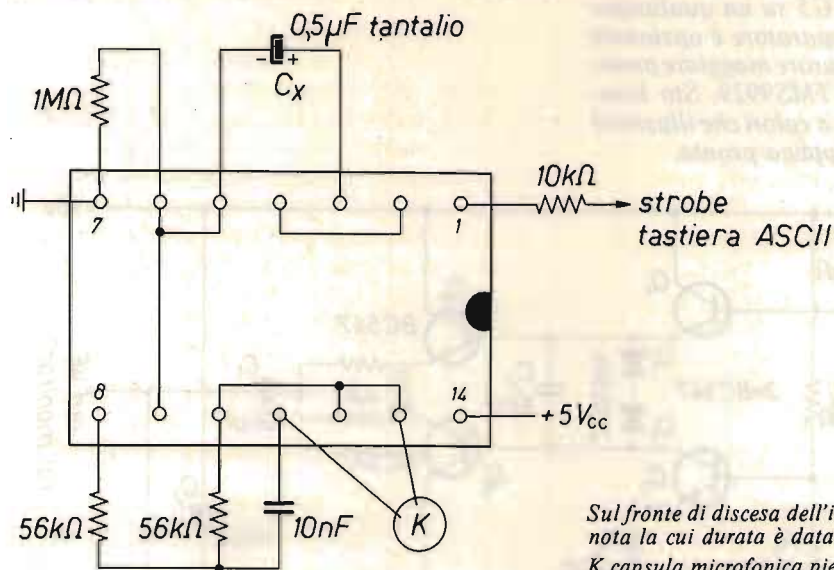
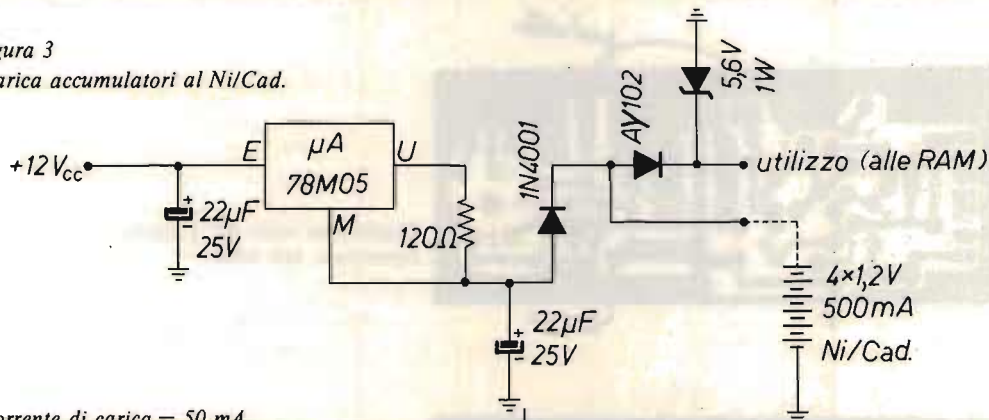



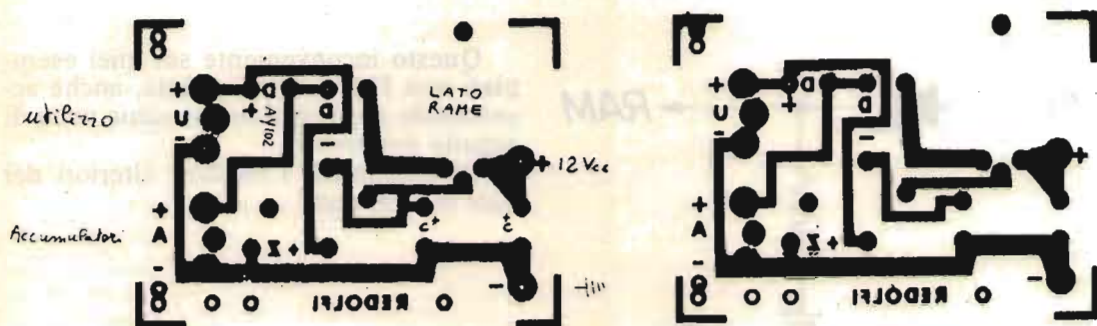
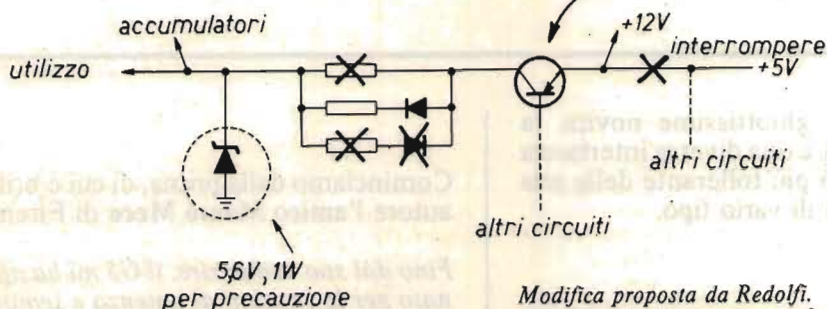
figura 2

Sul fronte di discesa dell'impulso di strobe emette una nota la cui durata è data da C_x.
K capsula microfonica piezoelettrica Ø 3,5 cm (GBC).

CARICA ACCUMULATORI *(vale solo per chi usa RAM C-MOS) - Il circuito di carica accumulatori da' qualche problema. Secondo il mio parere, si potrebbe modificare leggermente il circuito incriminato come da disegno allegato; mi riservo di provare tale mo-*

figura 3
Carica accumulatori al Ni/Cad.

Corrente di carica = 50 mA.
Commento: trattasi di un generatore di corrente costante che utilizza come elemento attivo un regolatore di tensione integrato tipo µA 78M05C. La resistenza da 120 Ω stabilisce la corrente di uscita a 50 mA. Il diodo 1N4001 evita che gli accumulatori possano scaricarsi attraverso il regolatore quando viene tolta l'alimentazione generale. Il diodo AY102, diodo al germanio utilizzato come damper nei TV, provoca una caduta di tensione di circa 0,25 V onde riportare la tensione al morsetto di utilizzo entro limiti tollerabili per le RAM utilizzate. Lo zener da 5,6 V è previsto come valvola di sicurezza: infatti, venendo a mancare gli accumulatori, si verrebbe a trovare sul carico utilizzatore l'intera tensione di alimentazione con le conseguenze del caso!
Il circuito utilizza degli accumulatori da 500 mA/ora, tensione nominale di utilizzo 1,2 V per elemento, tensione di massima carica 5,6 V. Tensione utilizzabile dal carico compresa tra 4,55 V e 5,35 V.
Il circuito soprastante andrebbe a sostituire l'originale montato sul G5 con l'accorgimento di non montare gli accumulatori sul circuito stampato denominato madre in quanto, in caso di perdite di elettrolita...

difica negli esemplari di prossima realizzazione. Per tagliare la testa al toro, ho installato nei G5 già realizzati il circuito del quale allego fotocopia, notando un notevole miglioramento della situazione. Per il mio esemplare di G5 ho già predisposto un carica accumulatori automatico il cui schema opportunamente adeguato e riveduto, è stato tratto da un vecchio numero di Elektor; esso provvede, dopo le modifiche, alla carica degli accumulatori non appena gli stessi arrivano a una tensione minima di carica pari a 4,4 V per un periodo di 14 ore e con una corrente costante di 48 mA (accumulatori da 500 mA/h); l'aggeggio sarà sempre in funzione (è sufficiente lasciare la spina inserita nella rete in quanto è alimentato da un trasformatore indipendente da 2 VA e, qualora si tolga la spina, un relè provvede a isolare gli accumulatori dal circuito di carica onde evitare la scarica degli stessi. Di questo intrigo possiedo il master del circuito stampato grande poco più di una cartolina (il trasformatore è da c.s., un modello venduto dalla GBC a poco prezzo), e vi assicuro che più funzionale e sicuro di così non si poteva fare).

(segue figura 3)

Carica accumulatori originale

Modifica proposta da Redolfi.
Alimentando il transistor in figura con una tensione più elevata, la batteria di accumulatori arriverà senz'altro alla massima tensione di carica.

Su questo ultimo argomento vorrei aggiungere una nota. Premesso e riconosciuto che i circuiti analogici non mi sono mai piaciuti, accettai il consiglio di un amico che mi suggeriva come realizzare la sezione di carica accumulatori. In effetti, come afferma l'amico Redolfi, questo circuito non è una perla e da' luogo a inconvenienti minori (esplosione delle batterie per sovraccarico, reazioni nucleari, ecc.). L'inconveniente non è stato però molto sentito in quanto alla fin fine la quasi totalità degli utenti ha usato RAM non CMOS ponticellando l'alimentazione delle medesime e non installando la parte di ricerca. Ritengo tuttavia che la maniera migliore è quella riportata in figura 4.

RAM *(vale solo per chi usa RAM C-MOS) - Nota dolente! Sono difficili da reperire le C/MOS ma sembra che ultimamente il mercato si sia aperto e sono arrivate delle RCA a prezzi modici (6300 x 1).*

Ho notato purtroppo che molto spesso il gesto di accendere o spegnere il G5 provoca la modifica di parte del programma nelle stesse contenuto specialmente se è di una certa lunghezza. Potrei suggerire di pilotare gli ingressi di controllo o funzione delle 21C14 con porte del tipo "open collector" e resistenze di pull-up verso il +5 V_{cc}degli accumulatori; tale sistema potrebbe permettere di minimizzare il difetto consentendo di trarre un effettivo vantaggio dall'uso di tale tipo di memorie.

figura 4
Circuito proposto da Becattini.

Questo inconveniente sui miei esemplari non l'ho mai riscontrato, anche accendendo e spegnendo parecchie volte di seguito per prova.

Ci comunichi i risultati ulteriori dei suoi esperimenti.

Al signor Redolfi va il premio, offerta dalla ditta SUMUS di Firenze, una scheda T-PIO (interfaccia parallela 2 x 8 bit) per il G5.

Seguono due ghiottissime novità: la scheda 48K RAM e una diversa interfaccia cassette per il G5 più tollerante della mia verso registratori di vario tipo.

Cominciamo dalla prima, di cui è brillante autore l'amico **Mauro Meco** di Firenze:

Fino dal suo comparire, il G5 mi ha affascinato per le sue doti di potenza e semplicità, ma in particolare ho subito apprezzato, da buon "smanettone", la presenza, come sulle macchine "grandi", di un bus accessibile e di ben quattro posti scheda. Solo la capacità di

memoria trovavo insoddisfacente: non appena ho avuto disponibile il registratore ho rapidamente colmato i 4K e sentivo quindi la necessità di qualcosa di più grande. Ecco pertanto che, per non riempire i quattro posti scheda con altrettante schede da 4K, che in ogni caso mi avrebbero fornito solo 16K, ho progettato questa scheda da 48K che ho in funzione da lungo tempo e che risulta perfettamente corrispondente allo scopo.

La scheda

Nel progettare questa scheda ho adottato un trucchetto che solo pochi anni fa mi avrebbe fatto rabbrividire al solo pensiero: ho infatti "gettato via" integralmente ben 16K di RAM che sono presenti ma che non vengono utilizzati! Premesso infatti che sul G5 già sono utilizzati 16K per il GBASIC e che lo spazio di indirizzamento dello Z-80 è al massimo di 64K, per raggiungere i 48K avevo due strade: o utilizzare 3x8 = 24 chip 4116 (che fanno in tutto 48K esatti) o usare 8 chip 4164, che danno 64K, gettando via senza pietà i 16K di troppo. Questa seconda soluzione, da me scelta, presenta i seguenti vantaggi:

1) richiede un numero tre volte minore di integrati;

2) richiede un numero tre volte minore di zoccoli;

3) richiede una superficie di circuito stampato oltre che tre volte minore; il c.s. risulta inoltre più semplice e più facile da collaudare;

4) richiede una minore potenza di alimentazione;

5) richiede una sola tensione di alimentazione invece che tre;

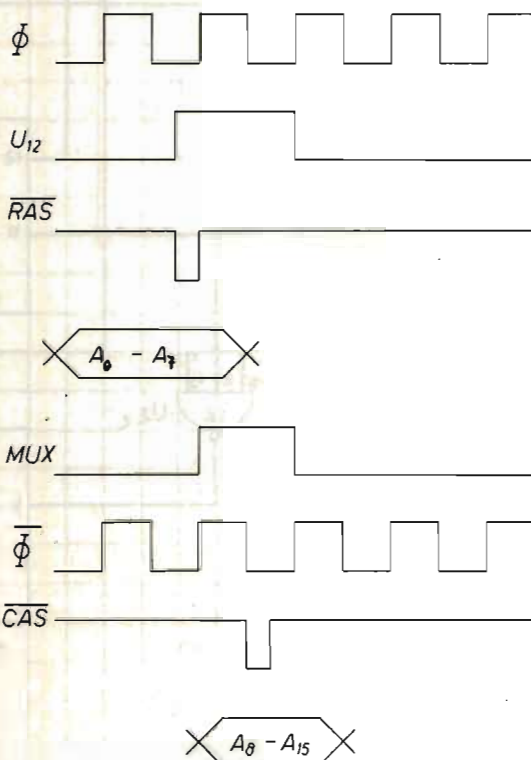
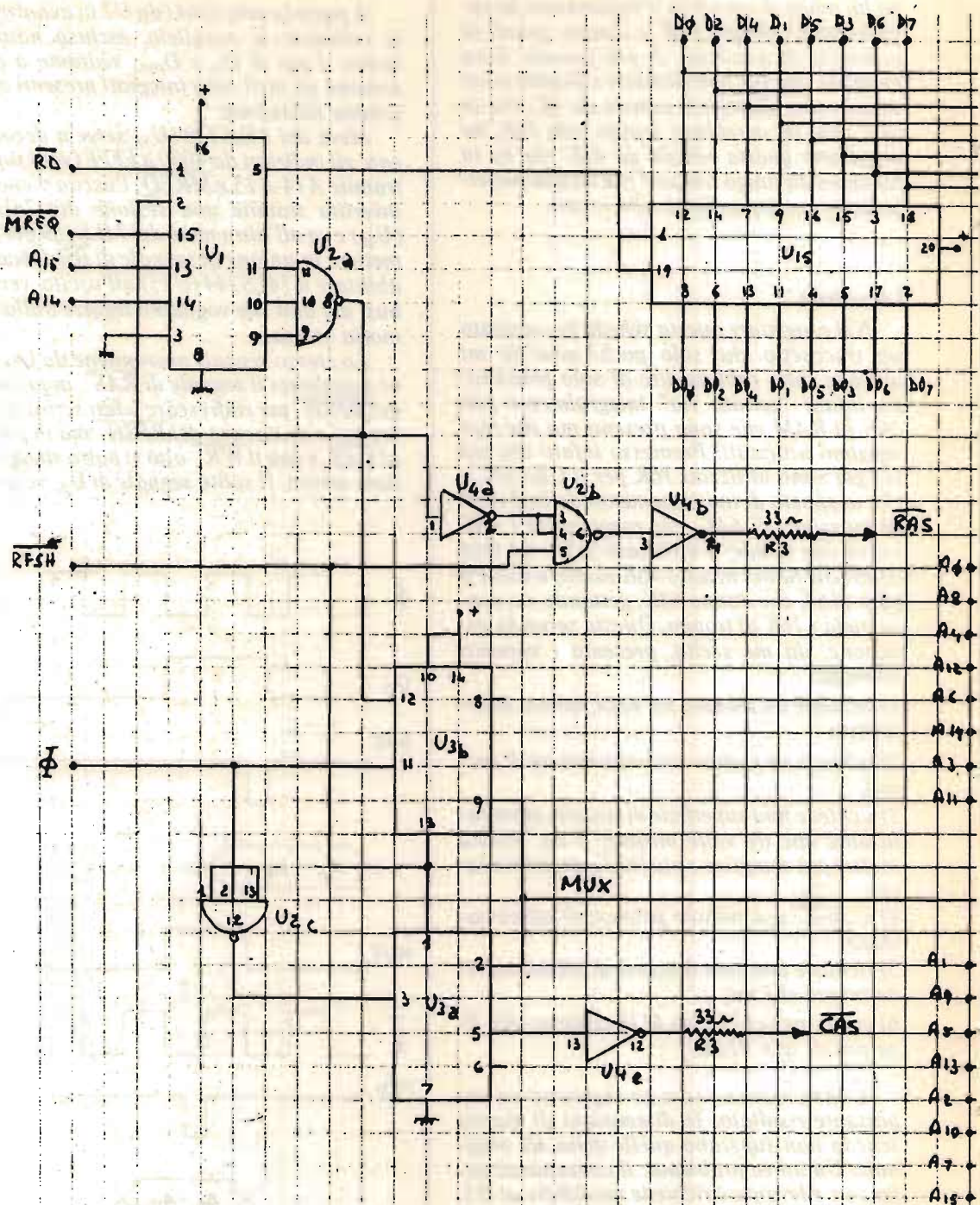
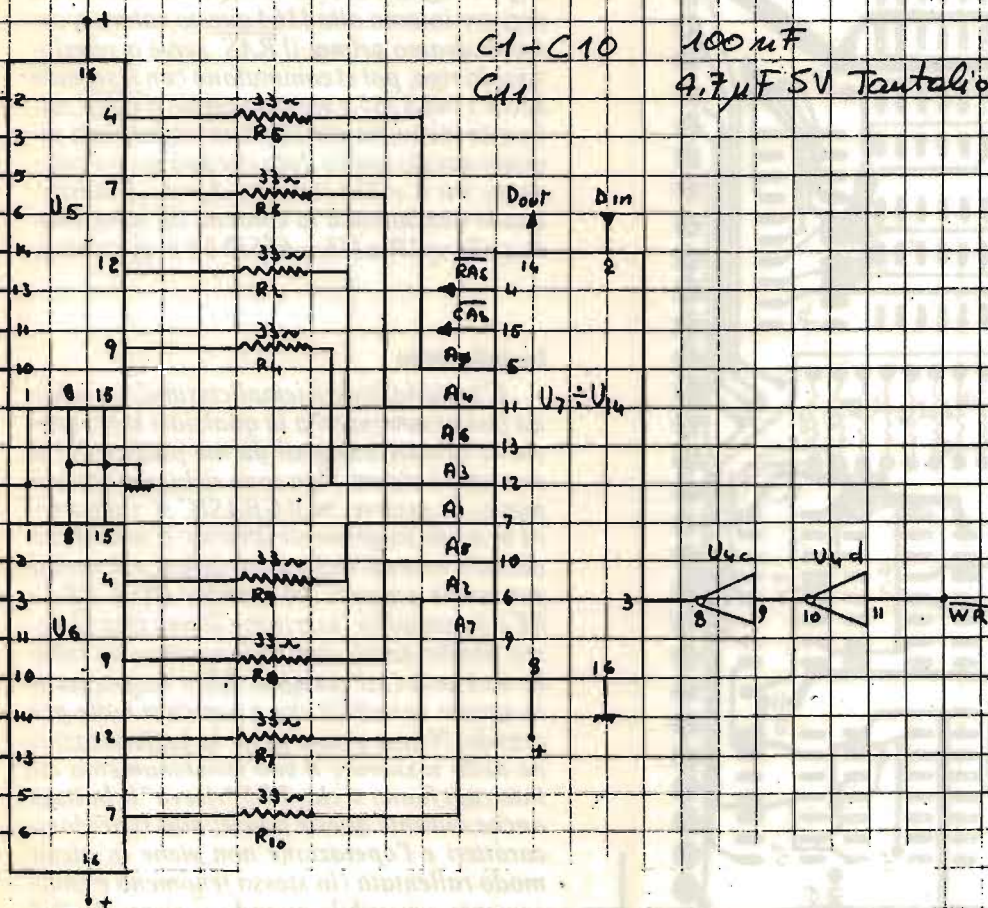
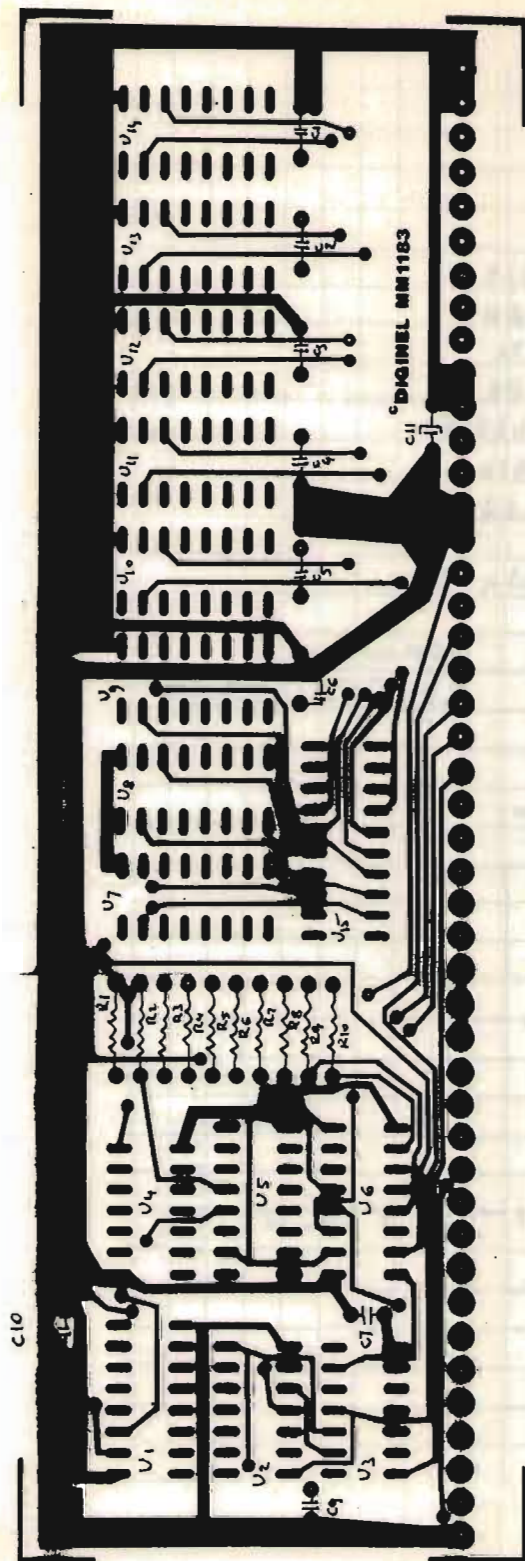
6) costa meno (24 chip 4116 costano alla fine più di otto 4116).

Con la tecnica usata ho raggiunto un importante risultato: le dimensioni di questa scheda non superano quelle della 4K originale. Un unico problema: il consumo di potenza è rilevante e richiede modifiche al G5. Si deve sostituire il 7805 con un 78H05. Il trasformatore inoltre deve avere il secondario che da'origine al 5 V sufficiente a fornire almeno una corrente di 3÷4 A.

Dettagli circuitali

A parte le otto 4164 (da U7 in avanti) tutte collegate in parallelo, escluso naturalmente il pin di D_{in} e D_{out}, vediamo a cosa servono gli altri sette integrati presenti sulla scheda medesima.

Metà del 74LS139 (U_1) serve a decodificare gli indirizzi da 4000 a FFFF (vedi sopra) tramite A14-A15 e $\overline{MREQ}$, l'uscita viene poi invertita tramite una sezione del 74LS10 (U_{2a}) e va all'altra metà del 74LS139 per permettere in unione al segnale di $\overline{RD}$ (read) di abilitare il 74LS244 (U15) all'uscita, verso il bus, dei dati che vogliamo leggere dalla memoria stessa.

Lo stesso segnale proveniente da U_{2a} serve a generare il segnale di $\overline{RAS}$, in presenza del $\overline{RFSH}$ per rinfrescare i dati scritti in memoria, e in assenza del $\overline{RFSH}$ ma in unione al $\overline{CAS}$ e con il $\overline{WR}$ alto si potrà rileggere il dato scritto. Il solito segnale di U_{2a} va anche

Temporizzazioni della scheda 48 k (Meco).

Schema 48k (Meco).

Disposizione dei componenti (Meco).

a una sezione di un 74LS74, esattamente U_{3b}, qui insieme al segnale di clock si genera un segnale attivo alto alla sua uscita Q che serve a commutare i due multiplexer (MUX) 74LS157 (U_5-U_6) tutto questo se non è presente lo $\overline{RFSH}$, infatti con $\overline{RFSH}$ attivo viene impedita la generazione del segnale MUX.

L'altra sezione dell'U_3 serve a generare il segnale $\overline{CAS}$.

Come dicevamo più sopra, gli indirizzi in ingresso della 4164 sono solo otto, ma allora si possono indirizzare solo 255 celle di memorie? No perché (vedi figura) quando è presente il $\overline{RAS}$ alle memorie viene presentato un indirizzo che va da 0 a 255 (tramite $A_0 \div A_7$) e il segnale di $\overline{RAS}$ serve a caricare in un registro interno alle 4164 questo valore e, come dicevamo prima, il $\overline{RAS}$ serve a selezionare la riga, poi si commutano con il segnale MUX i 74LS157 e si rende attivo il $\overline{CAS}$; allora la memoria caricherà in un secondo registro questo nuovo dato che può essere compreso tra 0 e 255 (tramite $A_8 \div A_{15}$) indirizzando questa volta la colonna da selezionare e allora 256x256 = 65535 bit indirizzabili.

Installazione

L'installazione è semplicissima. La scheda può essere inserita in qualsiasi slot e grazie ai circuiti stampati da me predisposti il risultato è sicuro. Non sono richieste messe a punto. Si osservi che il GBASIC si comporterà in modo leggermente diverso: si noterà che alcuni comandi richiedono infatti più tempo per essere eseguiti. Ad esempio RUN, ERASE o la semplice inserzione di una riga. Questo perché questi comandi operano su tutta la memoria (ad esempio, RUN azzera tutto lo spazio variabili) che è ora dieci volte più grande. Niente paura però: la bufferizzazione della tastiera e il suo funzionamento ad interrupt fanno si che il G5 "riceva" le battute anche durante queste pause; non si perdono caratteri e l'operazione non viene in alcun modo rallentata (lo stesso fenomeno è chiaramente avvertibile quando si risponde "N" al dialogo iniziale).

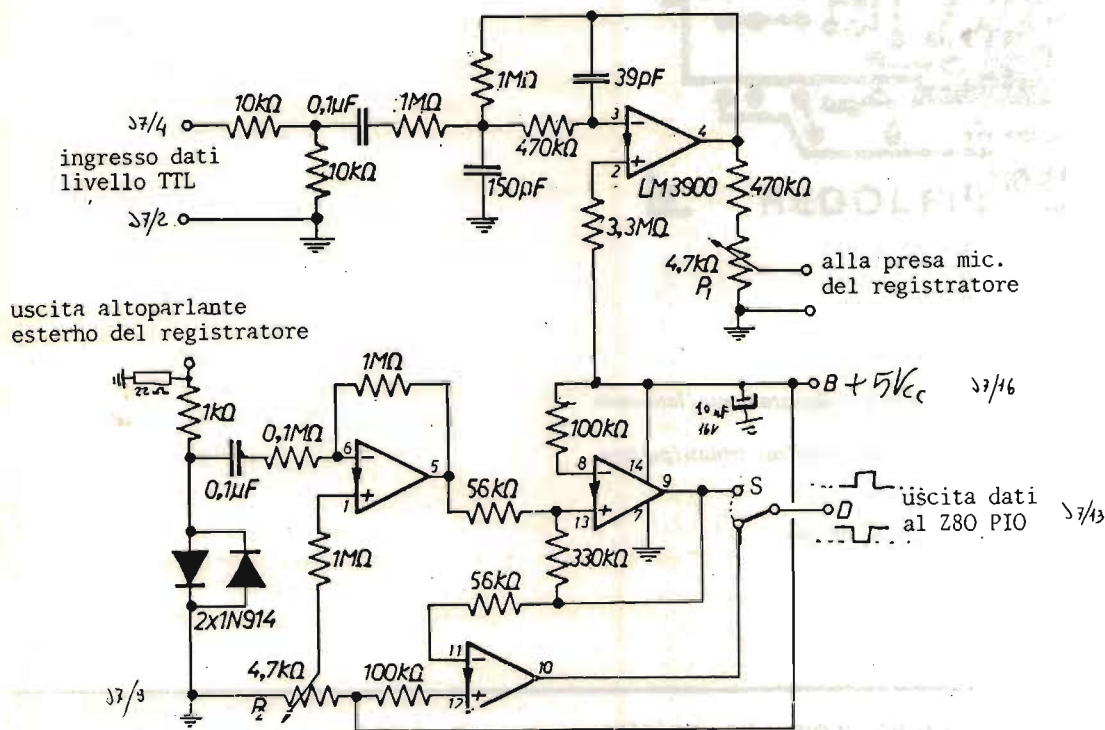
Secondo le usanze della rivista, elargisco al Meco materiale vario offerto nell'occasione dalla ditta SUMUS di Firenze che fra l'altro può fornire il c.s. della 48K.

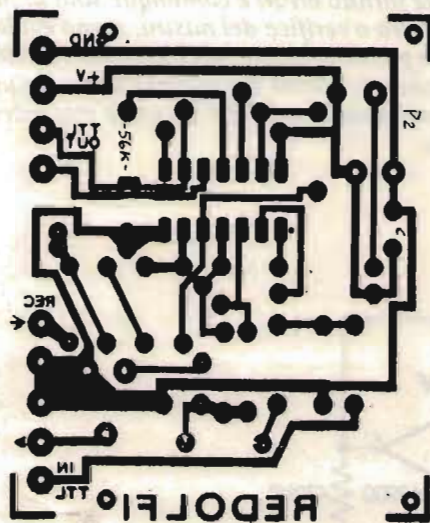
Cediamo ora la parola all'attivissimo **Claudio Redolfi** (via Moraro 26 - Monselice (PD)) che ci presenta una interessante interfaccia per cassette.

Prendendo lo spunto da un articolo di **CQ**, *invio lo schema di una interfaccia cassette da me realizzata per il G5 che utilizzo con buoni risultati con registratore Philips N2204 che, essendo dotato di un efficace CAG, si è dimostrato all'altezza del caso.*

Le regolazioni consistono nel girare il trimmer P_2 fino a leggere 2,5 V_{cc} sul pin 5 di IC e il P_2 a metà corsa. Con registratori non dotati di CAG è bene verificare di non saturare il nastro con segnali troppo forti onde evitare un notevole appiattimento dei picchi registrati.

In riproduzione, il livello di uscita del registratore deve essere posizionato in modo tale da consentire un fievole ascolto in altoparlante onde non amplificare un po' troppo il rumore di fondo.

Alla prova del nove, l'aggeggio ha raramente fornito errori e comunque solo in fase di lettura o verifica del nastro, segno evidente che possono essere raccolti rumori elettrici dai circuiti interni del registratore o da una commutazione in PLAY eseguita con indecisione.

Schema elettrico interfaccia registratore (Redolfi).

Il circuito è inseribile comodamente nel mobile del G5, nel mio caso sarà parte integrante di un registratore che stò realizzando con la mia solita caparbietà e che, qualora mi arrivi la scheda relativa a un ampliamento delle porte parallele di ingresso uscita da Lei promessami in una Sua precedente nota, sarà controllabile direttamente dalla tastiera del G5.

Interfaccia registratore. Circuito stampato, lato rame (Redolfi).

N.B. - Il trimmer P_1 ha i fori un po' ravvicinati (per errore).

Al Redolfi andrà in omaggio una interfaccia seriale per G5.

FINE

dottor Livio Andrea Bari

In questo periodo molti di noi radioamatori e inguaribili hardwaristi, sconvolti dal crollo dei prezzi dei calcolatori "personal", si sono fatti tentare e hanno acquistato uno ZX81 o uno Spectrum.
Io ho acquistato lo ZX81 senza alimentatore perché Danilo Risso mi aveva avvertito che dentro al power supply (alimentatore) Sinclair c'era molto poco in termini di componenti.
Lo stesso discorso vale per l'alimentatore da 9 V, 1,4 A destinato allo Spectrum.
Per alimentare in modo "professionale" questi personal computer della Sinclair ho realizzato per me e per alcuni amici diversi prototipi.

DIVERSE SOLUZIONI «PROFESSIONALI»

PER ALIMENTARE I COMPUTER ZX81, SPECTRUM E ACCESSORI (STAMPANTI, ECC.)

Perché conviene realizzare da sè l'alimentatore?

Per ottenere i seguenti vantaggi:

1) La tensione d'uscita dei nostri alimentatori è una vera corrente continua praticamente priva di ripple (ronzìo o residuo di alternata) mentre gli alimentatori originali Sinclair presentano qualche volt picco-picco di ripple.

2) Essendo la tensione d'uscita dei nostri alimentatori stabilizzata, il regolatore di tensione "on board" intorno al calcolatore (un 7805 in contenitore plastico) funziona con circa 3,5 V fissi tra i terminali di ingresso e d'uscita, pertanto stabilizza i 5 V necessari ai circuiti integrati interni in modo perfetto e in particolare dissipa meno potenza in calore e quindi scalda molto meno che col power supply originale: potete far funzionare il computer molto più a lungo senza dover temere danni da surriscaldamento.
3) Se l'ENEL vi rifila una tensione di rete inferiore alla nominale (ad esempio 190 V invece che 220 V) questo fatto non pregiudica il funzionamento del calcolatore.
4) I disturbi presenti sulla rete elettrica non raggiungono il calcolatore.
5) In tutti i nostri schemi è presente un pulsante di reset che, per dirla con Risso, serve a "sbloccare i vostri loop meno felici".

Sono previste tre opzioni per il gruppo trasformatore raddrizzatore (figure 1 e 2) e tre per lo stabilizzatore di tensione (figure 3, 4 e 5).

L'unione di uno dei tre possibili gruppi trasformatore-raddrizzatore con uno dei tre stabilizzatori permette di ottenere un alimentatore da rete stabilizzato. Chi possiede già un alimentatore con tensione di uscita in corrente continua di 12÷15 V con

figura 1

i	*interruttore a levetta 250 V, 1 A*
*C**	*condensatore elettrolitico da 4.700 μF, 25 V_L*
T	*trasformatore da 30 VA con secondario da 12 V*
P_D	*ponte a diodi da 200 V, 3 A, sostituibile da quattro diodi 1N5402 o BY251 o equivalenti connessi come da schizzo a lato:*

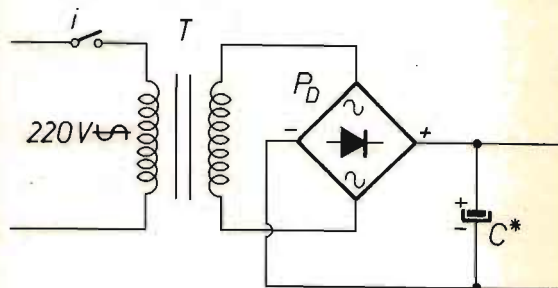
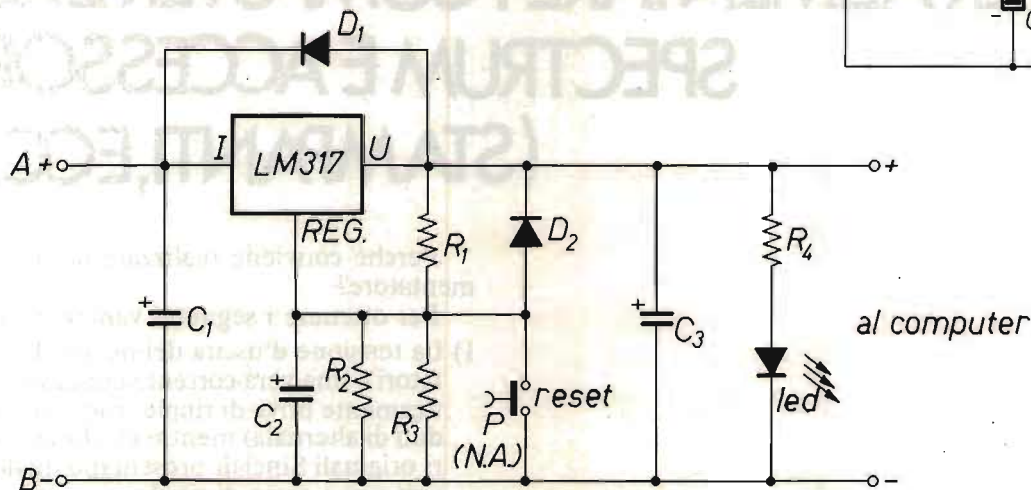
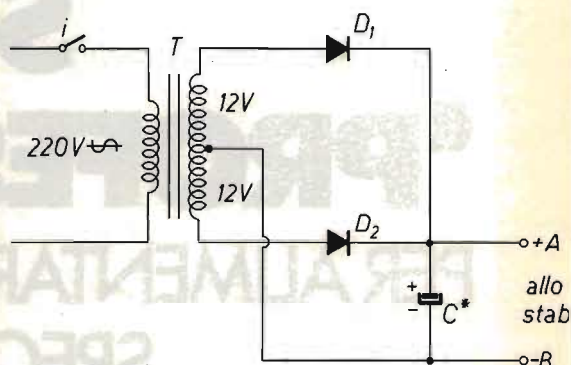
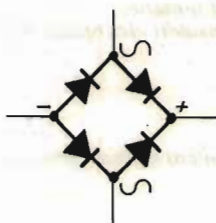



figura 2

T	*trasformatore con primario 220 V, secondario a presa centrale 12 V ÷ 0 ÷ 12 V, 1,5 A*
D_1, D_2	*diodi da almeno 3 A (ad esempio BY251 o 1N5402)*
*C**	*4.700 μF, 25 V_L, elettrolitico*

corrente non inferiore a 1,5 A può utilizzarlo in sostituzione del gruppo trasformatore-raddrizzatore.

In figura 3 vediamo uno stabilizzatore che usa il famoso regolatore di tensione LM317T.

Lo schema è leggermente più complesso di quello usato normalmente per alcuni accorgimenti (protezioni) che ne aumentano l'affidabilità.

A causa delle inevitabili tolleranze di costruzione dell'integrato LM317T e delle resistenze è necessario procedere a una taratura della tensione d'uscita che può essere compresa tra 8 e 9 V.
Costruito il circuito senza collegare R_3, si misura la tensione d'uscita stabilizzata che risulterà superiore a 9 V e si collega R_3 (da scegliere tra i valori 8,2; 10; 12; 15; 18 kΩ) in modo da ottenere una tensione compresa tra 8 e 9 V.

Io ho inserito $R_3 = 10$ kΩ per una tensione d'uscita di 8,2 V e lo ZX81 funziona ottimamente.

Se non trovate in vendita lo LM317T potete costruire lo stabilizzatore di figura 4 con un 7805 o quello di figura 5 con un 7808.

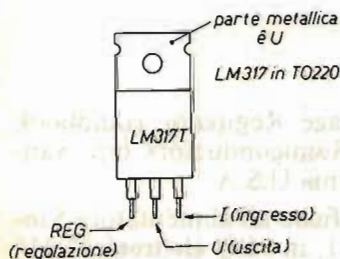
Per quanto concerne lo schema di figura 4, io ho ottenuto una tensione d'uscita di 8,5 V con $R_3 = 1$ kΩ.

La tensione può essere aggiustata modificando il valore di R_3.

Lo schema di figura 5 non necèssita di regolazioni ed è ancora più semplice.

I circuiti integrati regolatori di tensione

figura 3

D_1, D_2	*diodi al silicio 1 A, 200 V, ad esempio 1N4002*
C_1, C_2, C_3	*condensatori al tantalio 10 μF, 20 V (possono essere usati anche elettrolitici da 47 μF)*
R_1	*120 Ω*
R_2	*680 Ω*
R_3	*da scegliersi in sede di collaudo tra 8,2; 10; 12; 15; 18 kΩ oppure non inserita*
R_4	*680 Ω*
P	*pulsante con contatti N.A. (normalmente aperti)*

vanno montati su un dissipatore di calore o su una parete del contenitore se questo è di alluminio spesso almeno 1,5 mm. Non dimenticate di spalmare un poco di grasso al silicone sulla superficie di contatto tra regolatore e dissipatore.

Nello ZX81 lo spinotto maschio dell'alimentazione è un jack da 3,5 mm e il positivo deve essere collegato al centro.

figura 4

C_1, C_2	*10 µF, 20 V, al tantalio (oppure condensatori elettrolitici da 47 µF)*
R_1	*150 Ω*
R_2	*100 Ω*
R_3	*1 kΩ*
R_4	*680 Ω*
P	*pulsante con contatti normalmente chiusi (N.C.)*

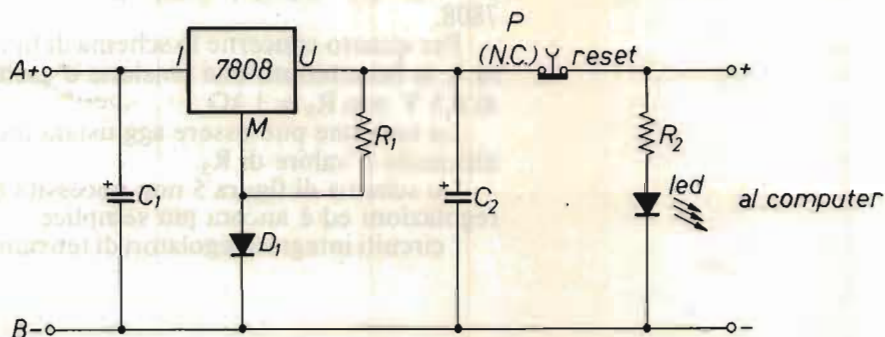
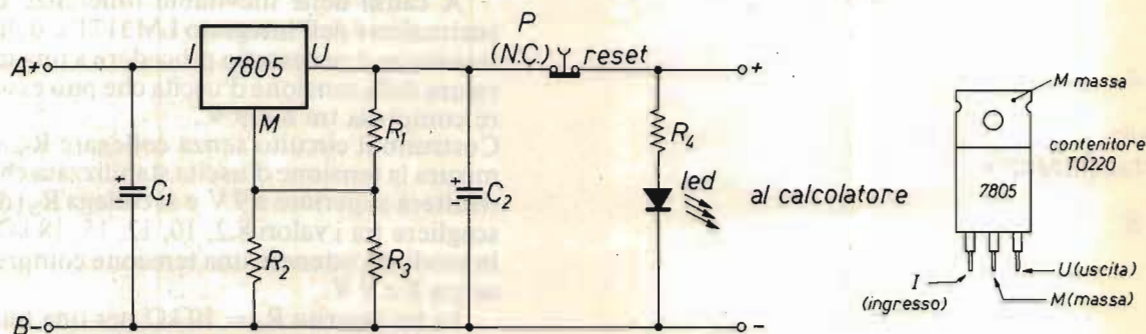



figura 5

C_1, C_2	*come in figura 4*
D_1	*diodo al silicio qualsiasi*
R_1	*820 Ω*
R_2	*680 Ω*

NOTA: le connessioni del 7808 sono come quelle del 7805

Bibliografia

AAVV, Voltage Regulator Handbook 1980, National Semiconductor Corp., Santa Clara California U.S.A.

Risso D., Modifiche all'alimentatore Sinclair dello ZX81, in **"CQ elettronica"** 11/1983.

FINE

p.e. Giancarlo Pisano

**Servendosi di un Personal Computer con adatto software, è possibile progettare molto semplicemente filtri del tipo passa-alto o passa-basso il cui utilizzo è molto ricorrente nel campo dell'alta frequenza.
Con il programma presentato in queste pagine si possono progettare tre tipi di filtri passa-basso e due tipi di passa-alto.**

RISOLUZIONE DI FILTRI

CON IL PERSONAL COMPUTER

Il programma è stato strutturato in modo molto semplice, utilizzando le istruzioni Basic più comuni (PRINT,INPUT,LET,IF-THEN,ecc...); grazie a ciò il programma può funzionare su qualsiasi microcomputer anche se la versione che presento è stata fatta girare con il noto TI99/4A.

L'unica istruzione "particolare" è la CALL CLEAR che serve per cancellare il video dopo aver dato il RUN. I possessori di computer diversi dal TI99, potranno comunque sostituire la CALL CLEAR con un'istruzione corrispondente, reperibile sul manuale d'uso dell'apparecchio.

Se anche l'istruzione END dovesse dar fastidio, provate a ometterla poiché il programma dovrebbe girare ugualmente ma ricordatevi di sostituire in modo appropriato i numeri di linea dopo i vari "THEN".

Guardiamoci il listato senza ulteriori indugi:

```
  5  call clear
 IO  rem-filtri passa alto e passa basso-
 20  print "calcolo di filtri passa alto e passa basso"
 30  print "opzioni; per i passa-basso: pi-greco,a "T",
     pi-greco elaborato;
     per i passa-alto: a "T", pi-greco"
 40  print "scrivi rispettivamente: PG,TB,PGE,TA,PGA"
 50  input A$
 60  if A$= "PG" then I20
 70  if A$= "TB" then 240
 80  if A$= "PGE" then 350
 90  if A$= "TA"  then 500
I00  if A$= "PGA" then 6IO
IIO  rem -pi greco-
I20  print "inserire la frequenza di lavoro in MHZ"
I30  input F
I40  let L=52/(3.I4*F)
I50  let C=I000000/(6.28*52*F)
I60  print "il valore di L e""   ";L
I70  print "il valore di C e""   ";C
I80  print "ancora il pi-greco ? (S/N)"
I90  input Z$
200  if Z$= "S" then I20
2IO  if Z$= "N" then 40
```

```
220  END
230  rem - "T" -
240  print "inserire la frequenza di lavoro in MHZ"
250  input F
260  let L=52/(6.28*F)
270  let C=I000000/(3.I4*52*F)
280  print "il valore di L e""     ";L
290  print "il valore di C e""     ";C
300  print "ancora il passa-basso a "T" ? (S/N)"
3I0  input Z$
320  if Z$= "S" then 240
330  if Z$= "N" then 40
340  END
350  rem- pi-greco elaborato -
360  print "inserire la frequenza di lavoro in MHZ"
370  input F
380  let L=52 0.6/(6.28*F)
390  let Ca=270000/(3.I4*52*F)
400  let Cu=600000/(6.28*52*F)
4I0  print "il valore di L e""     ";L
420  print "il valore di Ca e""     ";Ca
430  print "il valore di Cu e""     ";Cu
440  print "ancora il pi-greco elaborato ? (S/N)"
450  input Z$
460  if Z$= "S" then 360
470  if Z$= "N" then 40
480  END
```

```
640  let L=52/(6.28*F)
650  let C=I000000/(I2.56*52*F)
660  print "il valore di L e""     ";L
670  print "il valore di C e""     ";C
680  print "ancora il passa-alto a pi-greco ? (S/N)"
690  input Z$
700  if Z$= "S" then 620
7I0  if Z$= "N" then 40
720  END
490  rem - "T" passa alto -
500  print "inserire la frequenza di lavoro in MHZ"
5I0  input F
520  let L=52/(I2.56*F)
530  let C=I000000/(6.28*52*F)
540  print "il valore di L e""     ";L
550  print "il valore di C e""     ";C
560  print "ancora il passa-alto a "T" ? (S/N)"
570  input Z$
580  if Z$= "S" then 500
590  if Z$= "N" then 40
600  END
6I0  rem - passa alto a pi-greco -
620  print "inserire la frequenza di lavoro in MHZ"
630  input F
```

Dopo aver caricato il programma si darà il solito RUN e, se tutto è in regola, il computer ci chiederà quale filtro vogliamo calcolare. Premeremo PG per il pi-greco, TB per il passa-basso a "T", PGE per il pi-greco elaborato, TA per il passa-alto a "T", PGA per il passa-alto a pi-greco. Quindi, dopo aver dato uno di questi comandi premeremo ENTER; a quel punto dovremo digitare la frequenza che interessa e una volta premuto ENTER otterremo tutti i dati necessari per realizzare il filtro.

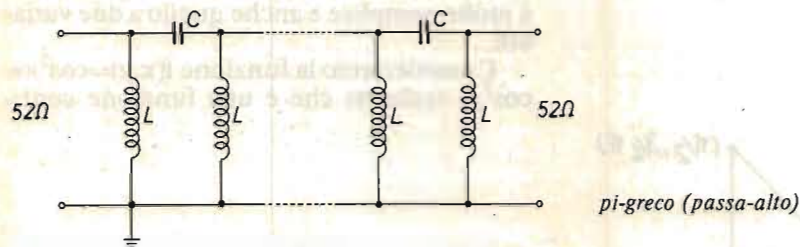
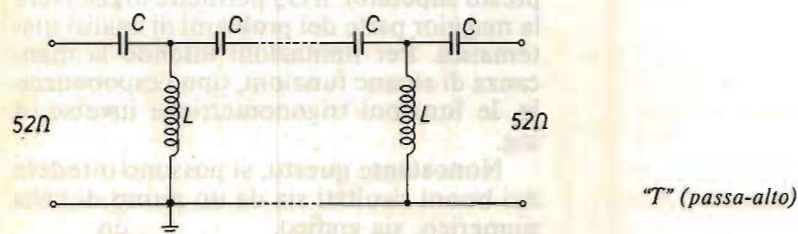
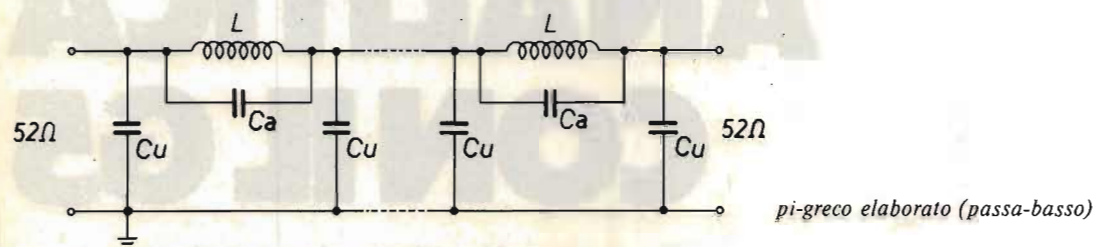
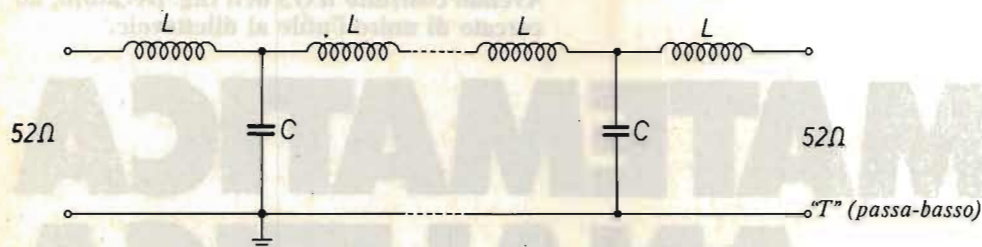
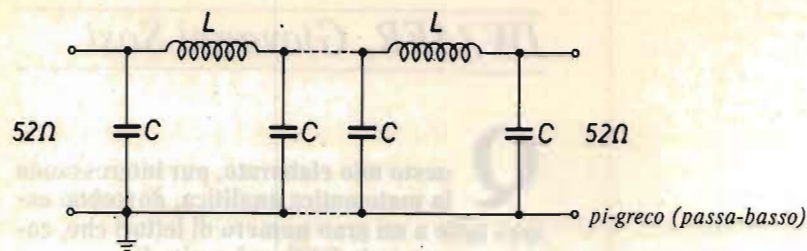
I filtri sono calcolati per avere un'impedenza di ingresso e di uscita uguale, pari a 52 Ω; se interessa un altro valore di impedenza, basta sostituire tale numero con quello adatto alle nostre esigenze.

Si tenga presente che collegando più filtri tra loro si ottiene una maggior attenuazione delle frequenze indesiderate ma è consigliabile non collegare tra loro più di tre filtri, per non attenuare anche la frequenza fondamentale.

Dovendo costruire un passa-alto, ricordiamoci di maggiorare leggermente il valore delle capacità. Per esempio, se il computer ci comunica che occorrono 14 pF, non utilizziamo un condensatore da 15 pF, ma un modello da 18 pF in quanto si correrebbe il rischio di attenuare anche la frequenza fondamentale.

Data l'utilità del programma è senza dubbio consigliabile salvarlo su cassetta (o analogo sistema del vostro microcomputer).

Non ci dovrebbero essere altri problemi, anche perché è il computer ad assumersi il compito più gravoso.

FINE

Il valore delle induttanze è espresso in microhenry, mentre il valore delle capacità è espresso in picofarad. Nel programma, l'istruzione END si può sostituire con STOP.

IW3AER, Giovanni Sosi

Questo mio elaborato, pur interessando la matematica analitica, dovrebbe essere utile a un gran numero di lettori che, come me, sono studenti universitari. Avendo costruito il G5 dell'Ing. Becattini, ho cercato di unire l'utile al dilettevole.

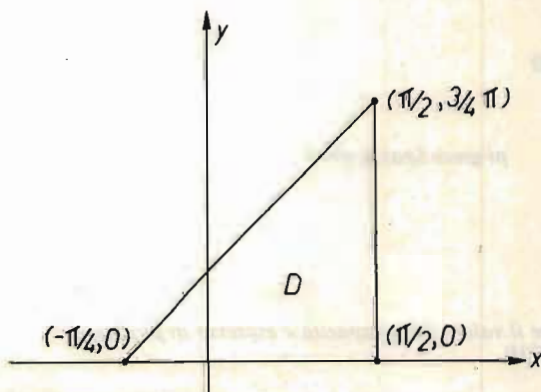
MATEMATICA ANALITICA CON IL G5

Nonostante alcune limitazioni (spero presto superate), il G5 permette di risolvere la maggior parte dei problemi di analisi matematica. Per limitazioni intendo la mancanza di alcune funzioni, tipo l'esponenziale, le funzioni trigonometriche inverse, il log.

Nonostante questo, si possono ottenere dei buoni risultati sia da un punto di vista numerico, sia grafico.

Lo studio di una funzione a una variabile è molto semplice e anche quello a due variabili.

Consideriamo la funzione $f(x,y)=\cos^2 x+\cos^2 y$; vediamo che è una funzione conti-

nua a due variabili e perciò a tre dimensioni.

Vogliamo trovare il massimo o il minimo di questa funzione entro un certo dominio D.

Volendo, oltre alla soluzione numerica, anche una soluzione grafica, dovrò decidere da che asse voglio "vedere" la mia curva; nel listato seguente è vista in direzione dell'asse y, ma nulla vieta di vederla dalle x o da ambedue, con la semplice sostituzione dell'istruzione SET...

```
5   REM MASSIMO E minimo DI F(X,Y)
IO  GRAPH
20 ORG I27,95
22 PLOT -I27,0,I27,0
24 PLOT 0,-95,0,95
26 FOR H=-I20 TO I20 STEP IO
28 PLOT H,-3,H,3
30 NEXT
32 FOR K=-90 TO 90 STEP IO
34 PLOT -3,K,3,K
36 NEXT
40 LET P=3.I4I5
45 LET MI=99999
46 LET M2=-99999
50 FOR X=-P/4 TO P/2 STEP 0.05
55 FOR Y=0 TO X+P/4 STEP 0.05
60 LET F=COS(X)xCOS(X)+COS(Y)xCOS(Y)
65 IF F<MI THEN LET MI=F
66 IF F<=MI THEN LET XI=X
67 IF F<=MI THEN LET YI=Y
70 IF F>M2 THEN LET M2=F
7I IF F>=M2 THEN LET X2=X
72 IF F>=M2 THEN LET Y2=Y
80 SET 40xX,40xF
82 NEXT
84 NEXT
86 AT I5,0
88 PRINT"minimo=",MI
90 PRINT"      X=",XI
92 PRINT"      Y=",YI
94 PRINT
96 PRINT"MASSIMO=",M2
98 PRINT"       X=",X2
99 PRINT"       Y=",Y2
I00 END
```

Naturalmente si possono aggiungere altre parti di completamento, per esempio tracciare il dominio; si possono definire due sprites a forma di freccia uno ↑ per il max e uno ↓ per il min. Questi vengono posizionati automaticamente sul tracciato del dominio per indicare automaticamente la posizione di massimo e di minimo. Insomma, chi più ne ha più ne metta.

FINE

I5YJI, Francesco Francescangeli

**"Questo Francescangeli non demorde!", dirà qualcuno.
A bere e bestemmiare tutto sta a cominciare, dice un detto toscano, uno dei tanti.
Scherzi a parte, sono affascinato dal G5.
La sorpresa dicembrina contenuta in XÉLECTRON, poi, è stata veramente gradita, portando le capacità del G5 a un livello decisamente interessante, se paragonato ad altre macchine commerciali.
Se poi penso alle future migliorie lasciate intravedere dall'Ing. Becattini...**

IO E IL G5

(release 3.00)

Per ora, comunque restiamo con i piedi in terra; passo quindi a farvi partecipi di una scoperta tanto inattesa quanto utilissima: il G5 ha capacità di EDITING, e oltretutto con una procedura semplicissima. Sperando vivamente che ci sia qualcuno a un livello di conoscenza più basso del mio, spiego brevemente: l'EDITING consiste nella possibilità di intervenire (con correzioni, cancellazioni e inserimenti) in un programma già steso, ma senza dover ribattere completamente la linea.

Ad esempio, se avete battuto

200 LET A=INT((N/16-INT5N/16))*15)+1

con il vecchio release (2.00) per correggere il +1 con un +2 avreste dovuto riscrivere tutta la linea. Con il nuovo release (3.00) invece la procedura è molto semplice:

1) fate il LIST del programma, o meglio solo della linea interessata con LIST 200,1
2) tramite i tasti di controllo del cursore (C+O=↑, C+J=↓, C+I=→, C+H (o Back Space)=←, C+N (o barra spazio)=→) posizionare quest'ultimo sotto l'1
3) battete 2

4) fate RETURN
Tutto qui.

L'uscita dal modo EDIT avviene automaticamente con il RETURN.

Se non credete che sia semplice, provate a farlo con un M20.

Per chiarezza di esposizione ritengo comunque che sia meglio esemplificare le varie possibilità.

CAMBIO DI UN DATO in mezzo a una linea

Fare le operazioni 1,2 e 3 di cui sopra, però attenzione: se battete il RETURN, tutto il resto della linea, pur restando sullo schermo, verrà cancellato dalla memoria; per ovviare a ciò basta portare il cursore in fondo alla linea con alcuni C+J. Quindi ricordatevi: tutto ciò che è dopo il cursore viene cancellato con il RETURN.

CANCELLAZIONE di parte di una linea

Sfruttando quanto detto sopra, basta portare il cursore sul primo carattere da cancellare e poi fare RETURN.

INSERIMENTO di dati

Portando il cursore in una certa posizione, tutto ciò che vi è dopo viene spostato in blocco con C+Z di tanti spazi quante sono le volte che si fa C+Z, fermo restando il cursore dove lo abbiamo posizionato; quindi, se vogliamo inserire una frase di 10 lettere, basta fare C+Z per 10 volte e, nello spazio libero, inserire la frase voluta. Ricordatevi sempre di portare il cursore in fondo alla linea prima del RETURN.

Mi sembra che sia alquanto utile, no?

E ora, tanto per cambiare, vorrei sottoporvi un programmino che ritengo sia utile per l'inserimento di programmi di linguaggio macchina, che il G5 accetta solo in decimale, o qualora ci fosse il bisogno di convertire in decimale lunghe file di esadecimali (almeno non fondete più la calcolatrice!); quando poi si parla di conversione da decimale a esadecimale, i più svengono e i sopravissuti affogano miseramente sotto tonnellate di foglietti coperti di divisioni e moltiplicazioni. Provare per credere.

Ora, dopo altrettante tonnellate di carta (una tantum) e dopo ore di DEBUG (ricerca errori: tanti!) sembra che sia riuscito a risolvere il problema.

Eccovi quindi il MEGAPROGRAMMA (tutto è relativo):

```
10 DIM H$(1)
20 PRINT CHR(12)
30 PRINT TAB(12),"MENU"
40 PRINT
50 PRINT
60 PRINT
70 REM - INSERIRE SEMPRE TUTTI E 4 I CARATTERI,
      ANCHE SE Ø
80 PRINT "1) CONVERSIONE HEXADECIMAL - DECIMALE"
90 PRINT
100 PRINT
110 PRINT "2) CONVERSIONE DECIMALE - HEXADECIMAL"
120 PRINT
130 PRINT
140 PRINT
150 PRINT TAB(12),"QUALE?":INPUT,H$
160 IF H$="1" THEN GOTO 180
170 IF H$="2" THEN GOTO 470
180 PRINT CHR(12)
190 DIM N$(1),Z$(1)
200 CLEAR
210 PRINT"PER FAVORE UN CARATTERE ALLA VOLTA"
220 PRINT
230 PRINT
240 FOR I=Ø TO 3
250 PRINT "CARATTERE?":INFUT,N$
260 GOSUB 400
270 ON I GOTO 310,300,290,280
280 LET N1=A:GOTO 320
290 LET N1=A*16:GOTO 320
300 LET N1=A*16*16:GOTO 320
310 LET N1=A*16*16*16:GOTO 320
320 LET N2=N1+N2
330 NEXT
```

```
340 PRINT N2
350 PRINT
360 PRINT
370 PRINT "VUOI RIPETERE? (S/N)":INPUT,Z$
380 IF Z$="S" THEN GOTO 200
390 IF Z$="N" THEN END
400 IF N$="A" THEN LET A=10:RETURN
410 IF N$="B" THEN LET A=11:RETURN
420 IF N$="C" THEN LET A=12:RETURN
430 IF N$="D" THEN LET A=13:RETURN
440 IF N$="E" THEN LET A=14:RETURN
450 IF N$="F" THEN LET A=15:RETURN
460 IF VAL(N$)<10 THEN LET A=VAL(N$):RETURN
470 PRINT CHR(12)
480 DIM A$(1),B$(1),C$(1),D$(1),X$(1),Y$(1)
490 CLEAR
500 PRINT "VALORE DA CONVERTIRE?":INPUT,N
510 GOSUB 760
520 IF X>9 THEN GOSUB 800
530 IF X<10 THEN LET X$=STR$(X)
540 LET D$=X$
550 LET N=Y
560 GOSUB 760
570 IF X>9 THEN GOSUB 800
580 IF X<10 THEN LET X$=STR$(X)
590 LET C$=X$
600 LET N=Y
610 GOSUB 760
620 IF X>9 THEN GOSUB 800
630 IF X<10 THEN LET X$=STR$(X)
640 LET B$=X$
650 LET N=Y
660 GOSUB 760
670 IF X>9 THEN GOSUB 800
680 IF X<10 THEN LET X$=STR$(X)
690 LET A$=X$
700 PRINT A$,B$,C$,D$
710 PRINT
720 PRINT
730 PRINT "VUOI RIPETERE? (S/N)":INPUT,Y$
740 IF Y$="S" THEN GOTO 490
750 IF Y$="N" THEN END
760 LET X=INT((N/16-INT(N/16))*15)
770 IF N/16-INT(N/16)<>Ø THEN LET X=X+1
780 LET Y=INT(N/16)
790 RETURN
800 IF X=10 THEN LET X$="A":RETURN
810 IF X=11 THEN LET X$="B":RETURN
820 IF X=12 THEN LET X$="C":RETURN
830 IF X=13 THEN LET X$="D":RETURN
840 IF X=14 THEN LET X$="E":RETURN
850 IF X=15 THEN LET X$="F":RETURN
```

E ora alcune osservazioni sul nuovo firmware del "nostro".

INPUT

La versione 3.00 non accetta più la frase INPUT N, pena segnalazione ERR INPUT, ma bisogna scrivere INPUT,N. Si viene però a perdere la, qualche volta, utile indicazione del ?_

Non viene accettata neanche la frase INPUT"mm",N ma bisogna scrivere, alla ZX81,

```
10 PRINT"mm"
20 INPUT,N
```

Per le stringhe attenzione: se scrivete INPUT G$ appare, si, il ? ma viene interpretato come facente parte della stringa; immettendo cioè la lettera N, G$ risulterà ?N. E allora? Come sopra INPUT,G$ e attesa senza il ?, oppure

```
10 PRINT"mm"
20 INPUT G$
```

Attenzione a non scrivere INPUT"mm",G$ o il "mm" farà parte della stringa. Provate:

```
5 DIM G$(10)
10 INPUT "NUMERO?", G$
```

```
20 PRINT G$
30 GOTO 10
```

Visto? Sinceramente lo trovo un po' scomodo.

STRINGHE

A proposito delle stringhe, notare alla linea 30 di quanto sopra il GOTO 10 e non GOTO 5. Non fate mai un ritorno a un DIMn$(x), pena ERR DIMENSION. Anche logico: una volta dimensionato, che resta da dimensionare? O no? Mah!

In compenso di quanto sopra, però, il G5 ci fornisce la non poco apprezzabile possibilità di scrivere con caratteri a doppia altezza, o da programma, con PRINT CHR(22) per abilitarla e PRINT CHR(23) per disabilitarla, o direttamente con CONTROL+V (CHR(22) e CONTROL+W (CHR(23).

Questo è molto più comodo, in quanto con altre macchine occorre un programma apposito per realizzare quanto sopra. Se poi ci fosse una istruzione tipo COPY, per travasare direttamente il contenuto dello schermo su stampante...

Dopo tanto stress, una divagazione amena.
Con la frase CHR$(x) si dà valore di stringa a uno dei caratteri del sistema, intendendo per carattere i segni, la punteggiatura, i numeri e le lettere.

Cioè con:

```
10 DIM C$(Ø)
20 LET C$=STR$(69)
30 PRINT C$
```

vi verrà stampata una E.

Dato che la successione dei caratteri è sempre la solita, vedi articolo precedente, il codice dei caratteri parte da 32 (Space), 33 (!) e via via sino al 126 (˜) e 127 (▦).

A titolo di cronaca vi informo che se eseguite un programmino del genere:

```
 5 DIM A$(1),B$(1)
10 LET A$=CHR$(65)
20 LET B$=CHR$(47)
30 IF A$>B$ THEN PRINT "OK":END
40 PRINT "NO"
```

vedrete che, indipendentemente dal fatto che il CHR$ sia lettera, numero o carattere, più è alta la sua posizione e più vale, suppongo perché il G5 riconosce il valore binario del carattere, che va a salire.

Logica conseguenza, nelle stringhe anche "BOC">"BOB", "TOM">"BIM" e "KKØ"<"AEB". A che può servire? In un ipotetico archivio a catalogare dati o nomi in ordine numerico o alfabetico o, meno seriamente, a vedere se valgo più io di mia moglie, numericamente parlando.

A questo punto è di prammatica presentare le nuove funzioni ottenibili con il famigerato tasto CONTROL.

Però stavolta, per non confondervi le idee, avrei pensato di presentare sotto forma di tabella, utile anche per tenerla sotto mano (non strappate **XÉLECTRON**, fotocopiatelo).

Il +R significa che occorre battere il RETURN.

CONTROL					
C +	TEXT		GRAPH		EDIT
	DIRETTO	PROGRAMMA	DIRETTO	PROGRAMMA	
A	cursore inizio linea	-	-	-	SI
B	-	-	INK	INK	-
C	STOP	-	STOP	-	-
D	riconoscimento minuscole	-	riconoscimento minuscole	-	-

E	GRAPH +R	GRAPH	GRAPH +R	GRAPH	-
F	BLANK +R	BLANK	BLANK +R	BLANK	-
G	NOBLANK +R	NOBLANK	NOBLANK +R	NOBLANK	-
H	←	-	←	-	SI
I	→ 8 spazi	-	-	-	SI
J	↓	-	↓	-	SI
K	cursore inizio pagina	-	-	-	SI
L	CLS	-			
M	RETURN	-			
N	→				SI
O	↑	-	-	-	SI
P	-	SPFORM	-	SPFORM	-
Q	-	SPDEF	-	SPDEF	-
R	-	SPTYP	-	SPTYP	-
S	-	AT	-	AT	
T	-	SAVE	-	SAVE	-
U	-	LOAD	-	LOAD	-
V	abilita sottolineatura	-	abilita ingrandimento	-	-
W	disabilita sottolineatura	-	disabilita ingrandimento	-	-
X	cancella riga in corso	-	-	-	SI
Z	fa spazio in linea	-	-	-	SI
VBW	-	IF	-	-	-
VEW	-	RETURN	-	-	-
VFW	-	READ	-	-	-

VGW	-	DATA	-	-	-
VPW	-	CLEAR	-	-	-
VQW	-	ARG	-	-	-
VRW	-	POKE	-	-	-
VSW	-	OUT	-	-	-
VTW	-	ON	-	-	-
VUW	-	STR$(	-	-	-
VØW	-	LET	-	-	-
V[W	-	CHR(	-	-	-
V]W	-	THEN	-	-	-
V\W	-	TAB	-	-	-
V←W	-	STEP	-	-	-
V↑W	-	TO	-	-	-
V RUB. W	-	INFO	-	-	-

Un programma senza pretese e con inganno, però, ve lo voglio rifilare.

```
10 PRINT CHR(12)
20 PRINT"INSERISCI L'ORA ESATTA IN ORE,MIN. E SEC."
30 INPUT,O,M,S
40 GOSUB 1000
50 FOR X=O TO 11
60 FOR Y=M TO 59
70 FOR Z=S TO 59
80 FOR A=Ø TO 410:NEXT
90 LET S=S+1
100 GOSUB 1000
110 NEXT Z
120 LET S=Ø
130 LET M=M+1
140 GOSUB 1000
150 NEXT Y
160 LET M=Ø
170 LET O=O+1
180 GOSUB 1000
190 NEXT X
200 LET O=Ø
210 GOTO 40
1000 PRINT CHR(12)
1010 PRINT O,M,S
1020 RETURN
```

L'inganno consiste nel fatto che non sono riuscito a far restare l'ora al suo posto durante lo svolgimento di un altro programma.

Se ci riuscite voi vi pago un caffè.

Se, anzi, avete qualche buon consiglio da darmi o qualche buon programma da offrirmi, vi ringrazio anticipatamente e spero di ricambiare.

Scrivetemi alla Casella Postale 20 - 58100 GROSSETO.

Anche per questa volta ho finito.

Spero comunque di rifarmi vivo presto, magari con qualcosa di hardware, tanto per cambiare argomento, che magari voi lo preferite!

Ringraziando quindi l'Ing. Becattini per l'impegno dimostrato nel far crescere la sua creatura in modo egregio, e complimentandomi con lui per la pregevole opera svolta, vi saluto tutti.

E non dite "meno male!", perché c'è una

ERRATA CORRIGE

Nel mio articolo su **XÉLECTRON** u.s., a pagina 77 nel programmino in alto manca la linea

20 FOR I=A TO B

Ciao.

FINE

Marco Bisaccioni

Normalmente, nella vita computeristica di ciascuno di noi, il basic risolve i nostri problemi, non però quando occorre avere particolare velocità di esecuzione.

MLX PER C64

Infatti ricordiamoci che ogni qualvolta chiediamo al nostro Home Computer di svolgere qualche istruzione in basic, esiste una serie di istruzioni allocata per lo più alla fine RAM che traducono in forma comprensibile al microprocessore, le suddette istruzioni.

Il basic, così come noi lo conosciamo, è infatti assolutamente incomprensibile per il microprocessore, che invece parla un altro idioma (non ridano gli addetti ai lavori) il così chiamato **"linguaggio macchina"**, cioè la lingua correntemente usata da quel tipo di microprocessore che è insediato nel nostro computer.

Quale che sia la macchina che noi possediamo, dentro di essa ha un microprocessore che a seconda del tipo, ha un suo set di istruzioni particolari.

Non vogliamo per ora addentrarci in questo discorso, ma sia chiaro che per forza esiste all'interno della macchina un programma che consente al computer di tradurre il, per lui incomprensibile, 'BASIC' in una serie di istruzioni in linguaggio macchina a lui invece comprensibili. È un po' come il diplomatico italiano che per capire il diplomatico arabo ha accanto a sè il traduttore che comprende l'arabo e lo traduce in italiano: senza l'interprete cercate di far parlare l'italiano con l'arabo, e il risultato se non altro porterebbe a un ennesimo aumento della benzina...

Logico, che se invece il diplomatico italiano conoscesse l'arabo, il loro colloquio sarebbe più veloce e personale. Così è per i nostri programmi in basic: il Sistema Operativo (appunto il programma traduttore) della macchina, deve, ogni qualvolta sia data una istruzione basic tradurla in linguaggio macchina: **si capisce pertanto la ragione per cui in basic il nostro computer perde velocità.** Poverino! Lui corre, ma pensate un po' a quello che deve fare: prelevare il comando basic, verificare se tale comando esiste nel suo sistema operativo e, se esiste, saltare alla routine che gli permette di svolgerla!

Tutti passaggi che fanno perdere tempo.

Ma allora se voglio un programma scheggiante come faccio?.

RISPOSTA: parlo con il microprocessore in linguaggio macchina.

Ma per far ciò occorre un programma che ci permetta di memorizzare nella macchina le nostre istruzioni in linguaggio macchina.

Questo programma si chiama MONITOR.

Il nostro fedele C64, attraverso il suo potente sistema operativo, ci permette di risolvere la maggior parte dei nostri "basicguai" con estrema facilità.

La cosa cambia drasticamente quando occorre lavorare in ML (Machine Language: linguaggio macchina): infatti il neo forse più grande del C64 è di non possedere un MONITOR. Certo, varie Ditte di software e la stessa Commodore hanno creato potenti monitor in grado di dare tutto l'aiuto possibile a chi intende creare un qualsiasi cosa in ML, ma in questo caso occorre mettere mano al porta...svalutazioni (...fogli!).

Ed ecco che per le cose non proprio difficili viene in nostro aiuto...:
MLX, letteralmente Machine Language Expanded è un programma che risolve il problema di caricare un programma in linguaggio macchina da basic.

MLX parte semplicemente caricandolo e poi digitando RUN.

Permette di salvare su disco o su nastro quanto creato e di caricarlo poi per l'utilizzo con le semplici e arciconosciute istruzioni:

LOAD "Programma", 1,1 (per cassetta)
LOAD "Programma", 8,1 (per disco)

Per lo start darete poi la SYS per la partenza del programma da voi creato.

Non appena caricato MLX, vi chiederà l'indirizzo di partenza da cui intendete partire e automaticamente presenterà una pagina testo ove d'ora in poi lavorerete.

UN EDITOR SPECIALE

Avete a disposizione un Editor che vi permette di sbagliare a iosa!

Con i comandi del C64 INST/DEL potete tornare indietro di una cifra alla volta, correggendola.

COMANDI DI CONTROLLO

S:SAVE, salva ciò che avete creato su nastro o disco tramite opzione.

L:LOAD, carica da nastro o disco un programma precedentemente creato per modificarlo oppure proseguirlo.

N:specifica Nuovo indirizzo, vi permette di spazzolare in avanti e indietro il programma attualmente in memoria.

D:Display, visualizza su schermo il contenuto di memoria che volete vedere da opzione.

Per rendere operabili questi comandi dovete semplicemente digitarli con lo Shift. Ad esempio: SHIFT/L = LOAD.

Troverete due listati: ambedue renderanno disponibile al vostro C64 il programma suddetto ma con due differenze.

MLX con cecksum di verifica: ogni linea è costituita da sette cifre l'ultima delle quali è appunto il così chiamato checksum che si calcola nel seguente modo. Il valore di linea +, i valori delle sei cifre costituenti la linea medesima, sommati tra di loro e ANDizzati con 255: ad esempio: 2049: 011 008 000 000 158 050 228

linea cf.1 cf.2 cf.3 cf.4 cf.5 cf.6 c.sum

Infatti 2049+((11+8+0+0+158+50)and255)=228

Provare per credere.

Logico che, in tutti quei casi in cui non è necessario creare un programma lungo, il checksum diventa noioso da calcolarsi e il lavoro che doveva essere corto diventa sicuramente più lungo.

È per questo che vi viene proposta anche una **versione modificata senza checksum di verifica.**

```
10 rem ##############################################################
20 rem #       m l x      con      cecksum      di      verifica     #
30 rem ##############################################################
100 print"";chr$(142);chr$(8);:poke53281,11:poke53280,14
101 rem  poke788,52
110 print"                                    ";
120 print"                                         ";
130 print"                                         ";
140 print"                                    "
142 print"machine code automatic editor"
145 print"        revision by xelectron bologna       "
150 v=53248:poke2040,13:poke2041,13:fori=832to894:pokei,255:next:pokev+27,3
160 pokev+21,3:pokev+39,2:pokev+40,2:pokev,144:pokev+1,54:pokev+2,192:pokev+3,54
170 pokev+29,3
180 fori=0to23:reada:poke679+i,a:pokev+39,a:pokev+40,a:next
185 data169,251,166,254,164,255,32,216,255,133,253,96
187 data169,0,166,251,164,252,32,213,255,133,253,96
190 pokev+39,5:pokev+40,5
210 print"mlx":print"indirizzo iniziale (dec) ";:inputs:f=1-f:c$=chr$(31+119*f)
220 ifs<256or(s>40960ands<49152)ors>53247thengosub3000:goto210
```

```
225 print:print:print
230 print"indirizzo finale      ";:inpute:f=1-f:c$=chr$(31+119*f)
240 ife<256or(e>40960ande<49152)ore>53247thengosub3000:goto230
250 ife<sthenprintc$;"█finale<start":gosub1000:goto230
260 print:print:print
300 print"█";chr$(14):ad=s:pokev+21,0
310 printright$("0000"+mid$(str$(ad),2),5);":";:forj=1to6
320 gosub570:ifn=-1thenj=j+n:goto320
390 ifn=-211then710
400 ifn=-204then790
410 ifn=-206thenprint:input"█nuovo indirizzo";zz
415 ifn=-206thenifzz<sorzz>ethenprint"█out of range█":gosub1000:goto410
417 ifn=-206thenad=zz:print:goto310
420 ifn<>-196then480
430 print:input"display da";f:print,"fino a";:inputt
440 iff<sorf>eort>ethenprint"almeno";s;"█, non piu' di";e:goto430
450 fori=ftotstep6:print:printright$("0000"+mid$(str$(i),2),5);":";
451 fork=0to5:n=peek(i+k):printright$("00"+mid$(str$(n),2),3);",";
460 geta$:ifa$>""thenprint:print:goto310
470 nextk:printchr$(20);:nexti:print:print:goto310
480 ifn<0thenprint:goto310
490 a(j)=n:nextj
500 cksum=ad-int(ad/256)*256:fori=1to6:cksum=(cksum+a(i))and255:next
510 printchr$(18);:gosub570:printchr$(20)
515 ifn=cksumthen530
520 print:print"linea sbagliata: ribattila":print:gosub1000:goto310
530 gosub2000
540 fori=1to6:pokead+i-1,a(i):next:poke54272,0:poke54273,0
550 ad=ad+6:ifad<ethen310
560 goto710
570 n=0:z=0
580 print"██ ██";
581 geta$:ifa$=""then581
585 printchr$(20);:a=asc(a$):ifa=13ora=44ora=32then670
590 ifa>128thenn=-a:return
600 ifa<>20then630
610 gosub690:ifi=1andt=44thenn=-1:print"██";:goto690
620 goto570
630 ifa<48ora>57then580
640 printa$;:n=n*10+a-48
650 ifn>255thena=20:gosub1000:goto600
660 z=z+1:ifz<3then580
670 ifz=0thengosub1000:goto570
680 print",";:return
690 s%=peek(209)+256*peek(210)+peek(211)
691 fori=1to3:t=peek(s%-i)
695 ift<>44andt<>58thenpokes%-i,32:next
700 printleft$("███",i-1);:return
710 print"██***save***███"
720 input"█file name";f$
730 print:print"███ape or ██isk: (t/d)"
740 geta$:ifa$<>"t"anda$<>"d"then740
750 dv=1-7*(a$="d"):ifdv=8thenf$="0:"+f$
760 open1,dv,1,f$:poke252,s/256:poke251,s-peek(252)*256
765 poke255,e/256:poke254,e-peek(255)*256
770 poke253,10:sys679:close1:ifpeek(253)>9orpeek(253)=0thenprint"█ok!":end
780 print"█errore di registrazione.  ritenta.":ifdv=1then720
781 open15,8,15:input#15,ds,ds$:printds;ds$:close15:goto720
790 print"██***load***██"
800 input"██file name";f$
810 print:print"███ape or ██isk: (t/d)"
820 geta$:ifa$<>"t"anda$<>"d"then820
830 dv=1-7*(a$="d"):ifdv=8thenf$="0:"+f$
840 open1,dv,0,f$:poke252,s/256:poke251,s-peek(252)*256
850 poke253,10:sys691:close1
860 ifpeek(253)>9orpeek(253)=0thenprint:print:goto310
```

```
870 Print"■Errore di Caricamento.  Ritenta.■":ifdv=1then800
880 oPen15,8,15:inPut#15,ds,ds$:Printds;ds$:close15:goto800
1000 rem*******buzzer********
1001 Poke54296,15:Poke54277,45:Poke54278,165
1002 Poke54276,33:Poke54273,6:Poke54272,5
1003 fort=1to200:next:Poke54276,32:Poke54273,0:Poke54272,0:return
2000 rem******bell sound*******
2001 Poke54296,15:Poke54277,0:Poke54278,247
2002 Poke54276,17:Poke54273,40:Poke54272,0
2003 fort=1to100:next:Poke54276,16:return
3000 Printc$;"■Pagina zero non accessibile":goto1000

ready.
```

MLX simple version: ogni linea è costituita naturalmente da sei cifre. Manca appunto l'ultimo numero check.

```
1 rem ##############################################################
2 rem #            m       l       x            simple version     #
3 rem ##############################################################
10 Print"■■";chr$(142);chr$(8);:Poke53281,11:Poke53280,14
14 rem   Poke788,52
18 Print"■                                          ";
22 Print"■                   ■■■■■ ■  ■■■■■                ";
26 Print"■                   ■ ■ ■ ■■ ■■■■■                ";
30 Print"■                                          "
34 Print"■■machine code automatic editor"
38 Print"■      revision by xelectron bologna      "
42 v=53248:Poke2040,13:Poke2041,13:fori=832to894:Pokei,255:next:Pokev+27,3
46 Pokev+21,3:Pokev+39,2:Pokev+40,2:Pokev,144:Pokev+1,54:Pokev+2,192
50 Pokev+3,55
54 Pokev+29,3
58 fori=0to23:reada:Poke679+i,a:Pokev+39,a:Pokev+40,a:next
62 data169,251,166,254,164,255,32,216,255,133,253,96
66 data169,0,166,251,164,252,32,213,255,133,253,96
70 Pokev+39,5:Pokev+40,5
74 Print"mlx":Print"indirizzo iniziale (dec) ";:inPuts:f=1-f:c$=chr$(31+119*f)
78 ifs<256or(s>40960ands<49152)ors>53247thengosub370:goto74
82 Print:Print:Print
86 Print"indirizzo finale          ";:inPute:f=1-f:c$=chr$(31+119*f)
90 ife<256or(e>40960ande<49152)ore>53247thengosub370:goto86
94 ife<sthenPrintc$;"■finale<start":gosub338:goto86
98 Print:Print:Print
102 Print"■";chr$(14):ad=s:Pokev+21,0
106 Printright$("0000"+mid$(str$(ad),2),5);":";:forj=1to6
110 gosub186:ifn=-1thenj=j+n:goto110
114 ifn=-211then258
118 ifn=-204then298
122 ifn=-206thenPrint:inPut"■nuovo indirizzo";zz
126 ifn=-206thenifzz<sorzz>ethenPrint"■out of range■:goS1000:go410
130 ifn=-206thenad=zz:Print:goto106
134 ifn<>-196then162
138 Print:inPut"display da";f:Print"   fino  a";:inPutt
142 iff<sorf>eort>ethenPrint"almeno";s;"■, non piu' di";e:goto138
146 fori=ftotstep6:Print:Printright$("0000"+mid$(str$(i),2),5);":",
150 fork=0to5:n=Peek(i+k):Printright$("00"+mid$(str$(n),2),3);",";
154 geta$:ifa$>""thenPrint:Print:goto106
158 nextk:Printchr$(20);:nexti:Print:Print:goto106
162 ifn<0thenPrint:goto106
166 a(j)=n:nextj
170 Print:gosub354
174 fori=1to6:Pokead+i-1,a(i):next:Poke54272,0:Poke54273,0
178 ad=ad+6:ifad<ethen106
182 goto258
```

```
186 n=0:z=0
190 print"█ █";
194 geta$:ifa$=""then194
198 printchr$(20);:a=asc(a$):ifa=13ora=44ora=32then234
202 ifa>128thenn=-a:return
206 ifa<>20then218
210 gosub242:ifi=1andt=44thenn=-1:print"██";:goto242
214 goto186
218 ifa<48ora>57then190
222 printa$;:n=n*10+a-48
226 ifn>255thena=20:gosub338:goto206
230 z=z+1:ifz<3then190
234 ifz=0thengosub338:goto186
238 print",";:return
242 s%=peek(209)+256*peek(210)+peek(211)
246 fori=1to3:t=peek(s%-i)
250 ift<>44andt<>58thenpokes%-i,32:next
254 printleft$("███",i-1);:return
258 print"██***Save***███"
262 input"█File name";f$
266 print:print"███Tape or █Disk: (T/D)"
270 geta$:ifa$<>"t"anda$<>"d"then270
274 dv=1-7*(a$="d"):ifdv=8thenf$="0:"+f$
278 open1,dv,1,f$:poke252,s/256:poke251,s-peek(252)*256
282 poke255,e/256:poke254,e-peek(255)*256
286 poke253,10:sys679:close1:ifpeek(253)>9orpeek(253)=0thenprint"█Fatto.":end
290 print"█Errore di Registrazione.  Ritenta.":ifdv=1then262
294 open15,8,15:input#15,ds,ds$:printds;ds$:close15:goto262
298 print"██***Load***██"
302 input"██File name";f$
306 print:print"███Tape or █Disk: (T/D)"
310 geta$:ifa$<>"t"anda$<>"d"then310
314 dv=1-7*(a$="d"):ifdv=8thenf$="0:"+f$
318 open1,dv,0,f$:poke252,s/256:poke251,s-peek(252)*256
322 poke253,10:sys691:close1
326 ifpeek(253)>9orpeek(253)=0thenprint:print:goto106
330 print"█Errore di Caricamento.  Ritenta.█":ifdv=1then302
334 open15,8,15:input#15,ds,ds$:printds;ds$:close15:goto302
338 rem*******buzzer*******
342 poke54296,15:poke54277,45:poke54278,165
346 poke54276,33:poke54273,6:poke54272,5
350 fort=1to200:next:poke54276,32:poke54273,0:poke54272,0:return
354 rem******bell sound******
358 poke54296,15:poke54277,0:poke54278,247
362 poke54276,17:poke54273,40:poke54272,0
366 fort=1to100:next:poke54276,16:return
370 printc$;"█Pagina Zero e Rom non accessibili":goto338

ready.
```

Il funzionamento è semplicissimo e lo diventerà ancor di più, come in tutte le cose, usandolo: usatelo, diventando esperti programmatori in ML per futuri programmi in XÉLECTRON XÉLEctron XÉlectron Xélectron xélectron!

FINE

IW1AYJ, Claudio Colombo

Ecco qui un programma per l'apprendimento del morse, che ha già dato ottimi risultati pratici, sottoscritto compreso. Chi ha intenzione di imparare il morse e possiede un VIC-20, non perda tempo.
Il primo programma permette di scrivere le 240 lettere della prova d'esame, dà la possibilità di correggere eventuali errori ed è in grado di trasmettere tutte le lettere in un secondo tempo alla velocità voluta.
Alla fine della trasmissione, il computer chiede se si vuole ritrasmettere la stessa pagina, anche a velocità diversa, oppure una pagina nuova.

COME DIVENTARE "SUPER" IN CW

OVVERO: «IMPARIAMO COL COMPUTER» DUE PROGRAMMI AD USO E CONSUMO DEI PIÙ BRAVI E DEI MENO BRAVI

Uno sguardo al programma

Il cuore del programma è rappresentato dalla linea 3 che genera il **punto** e dalla linea 4 che genera la **linea.** La linea 5 invece stabilisce la durata delle pause fra le lettere che compongono la pagina. Il numero che viene moltiplicato per "P" può variare da 1,5 a 3.

Tutti i dati riguardanti l'esecuzione delle lettere e dei numeri sono raggruppati dalla linea 48 alla linea 415.

La linea 40 fa iniziare il ciclo della trasmissione e la linea 43 stampa le lettere sul video, mentre vengono trasmesse, permettendo il controllo visivo.

La linea 650 calcola la durata del punto, unità di base che è funzione del numero

delle battute al minuto.

Se si tentasse di trasmettere a una velocità inferiore alle 40 battute al minuto, la lunghezza delle linee e dei punti verrebbe troppo dilatata nel tempo, rendendo difficile l'identificazione delle lettere.

È necessario, quindi, non scendere molto al di sotto di tale limite, senza apportare una modifica al programma, aggiungendo una nuova variabile, che si riferisca esclusivamente alla durata delle pause, lasciando invariata la durata del punto.

Per esempio, si potrebbe inserire questa piccola modifica:

```
600 INPUT K
610 IF K < 40 THEN GOTO 630
620 P=H=INT (3100)/K: GOTO 650
630 P=80
640 H=INT (3100)/K
650 L=3 * P
```

Si dovrà cambiare anche la linea 5

```
5 FOR T = 1 TO (3 * H): NEXT T:
  GOTO 420
```

E la linea 7000

```
7000 FOR T = 1 TO (2 * H): NEXT T:
     RETURN
```

```
1 PRINT"□":GOSUB1000:GOSUB3600
2 PRINT"□":GOSUB500:PRINT"□":GOTO10
3 POKES2,236:FORT=1TOP:NEXTT:POKES2,0:FORT=1TOP:NEXTT:RETURN
4 POKES2,236:FORT=1TOL:NEXTT:POKES2,0:FORT=1TOP:NEXTT:RETURN
5 FORT=1TO(3*P):NEXTT:GOTO420
10 S2=36876
12 POKE36878,10
13 PRINT"▮▮▮▮▮▮▮▮▮▮SCRIVI▮"
14 PRINT:PRINT
18 FORK=1TOX
22 GETA$(K):IFA$(K)=""THEN22
25 PRINTA$(K);
26 IFA$(K)=CHR$(20)THENK=(K-1):GOTO22
28 IFA$(K)="*"THENX=K :GOTO32
30 NEXTK
32 POKE36879,28
33 PRINT:PRINT
35 INPUT"PER INIZIARE BATTI ↑▮▮";M$
37 IFM$<>"↑"THEN33
38 PRINT"□"
40 FORK=1TOX
43 PRINT TAB(5)A$(K)
48 IFA$(K)="A"THENGOSUB3:GOSUB4:GOTO5
50 IFA$(K)="B"THENGOSUB4:GOSUB3:GOSUB3:GOSUB3:GOTO5
60 IFA$(K)="C"THENGOSUB4:GOSUB3:GOSUB4:GOSUB3:GOTO5
70 IFA$(K)="D"THENGOSUB4:GOSUB3:GOSUB3:GOTO5
80 IFA$(K)="E"THENGOSUB3:GOTO5
90 IFA$(K)="F"THEN GOSUB3:GOSUB3:GOSUB4:GOSUB3:GOTO5
100 IFA$(K)="G"THENGOSUB4:GOSUB4:GOSUB3:GOTO5
110 IFA$(K)="H"THENGOSUB3:GOSUB3:GOSUB3:GOSUB3:GOTO5
120 IFA$(K)="I"THENGOSUB3:GOSUB3:GOTO5
130 IFA$(K)="J"THENGOSUB3:    GOSUB4:GOSUB4:GOSUB4:GOTO5
140 IFA$(K)="K"THENGOSUB4:GOSUB3:GOSUB4:GOTO5
150 IFA$(K)="L"THENGOSUB3:GOSUB4:GOSUB3:GOSUB3:GOTO5
160 IFA$(K)="M"THENGOSUB4:GOSUB4:GOTO5
170 IFA$(K)="N"THENGOSUB4:GOSUB3:GOTO5
180 IFA$(K)="O"THENGOSUB4:GOSUB4:GOSUB4:GOTO5
190 IFA$(K)="P"THENGOSUB3:GOSUB4:GOSUB4:GOSUB3:GOTO5
```

```
200 IFA$(K)="Q"THENGOSUB4:GOSUB4:GOSUB3:GOSUB4:GOTO5
210 IFA$(K)="R"THENGOSUB3:GOSUB4:GOSUB3:GOTO5
220 IFA$(K)="S"THENGOSUB3:GOSUB3:GOSUB3:GOTO5
230 IFA$(K)="T"THENGOSUB4:GOTO5
240 IFA$(K)="U"THENGOSUB3:GOSUB3:GOSUB4:GOTO5
250 IFA$(K)="V"THENGOSUB3:GOSUB3:GOSUB3:GOSUB4:GOTO5
260 IFA$(K)="W"THENGOSUB3:GOSUB4:GOSUB4:GOTO5
270 IFA$(K)="X"THENGOSUB4:GOSUB3:GOSUB3:GOSUB4:GOTO5
280 IFA$(K)="Y"THENGOSUB4:GOSUB3:GOSUB4:GOSUB4:GOTO5
290 IFA$(K)="Z"THENGOSUB4:GOSUB4:GOSUB3:GOSUB3:GOTO5
300 IFA$(K)="1"THENGOSUB3:GOSUB4:GOSUB4:GOSUB4:GOSUB4:GOTO5
310 IFA$(K)="2"THENGOSUB3:GOSUB3:GOSUB4:GOSUB4:GOSUB4:GOTO5
320 IFA$(K)="3"THENGOSUB3:GOSUB3:GOSUB3:GOSUB4:GOSUB4:GOTO5
330 IFA$(K)="4"THENGOSUB3:GOSUB3:GOSUB3:GOSUB3:GOSUB4:GOTO5
340 IFA$(K)="5"THENGOSUB3:GOSUB3:GOSUB3:GOSUB3:GOSUB3:GOTO5
350 IFA$(K)="6"THENGOSUB4:    GOSUB3:GOSUB3:GOSUB3:GOSUB3:GOTO5
360 IFA$(K)="7"THENGOSUB4:GOSUB4:GOSUB3:GOSUB3:GOSUB3:GOTO5
370 IFA$(K)="8"THENGOSUB4:GOSUB4:GOSUB4:GOSUB3:GOSUB3:GOTO5
380 IFA$(K)="9"THENGOSUB4:GOSUB4:GOSUB4:GOSUB4:GOSUB3:GOTO5
390 IFA$(K)="0"THENGOSUB4:GOSUB4:GOSUB4:GOSUB4:GOSUB4:GOTO5
400 IFA$(K)=CHR$(32)THENGOSUB7000:GOTO5
415 IFA$(K)="@"THENGOTO8000
420 NEXTK
430 PRINT:PRINT
435 PRINT"ANCORA ?(S/N)"
440 INPUT C$:IFC$<>"S" AND C$<>"N"THEN440
445 IFC$="S"THENGOSUB500:GOTO38
450 IFC$="N"THENX=J:GOTO 2
500 POKE36879,29
510 PRINT"□"   :PRINT:PRINT:PRINT:PRINT
550 PRINT"QUANTE BATTUTE",
560 PRINT
570 PRINT"AL MINUTO";
600 INPUT K
650 P=INT(3100)/K
700 L=3*P
750 RETURN
1000 PRINT"*****CODIFICATORE*****",
1100 PRINT
1200 PRINT"         MORSE"
1250 PRINT
1300 PRINT"ISRUZIONI"
1350 PRINT
1400 PRINT"IL COMPIUTER PUO'"
1500 PRINT"REGISTRARE IN MEMORIA"
1600 PRINT"300 LETTERE E POI   "
1700 PRINT"TRASMETTERLE ALLA    "
1800 PRINT"VELOCITA' VOLUTA."
1900 PRINT"PER TRASMETTERE"
1950 PRINT"PREMI *"
2000 PRINT:PRINT
2100 PRINT"PER CACELLARE UN ER-"
2200 PRINT"RORE PREMI (DEL)"
```

```
2300 PRINT
2400 PRINT"PER FINIRE PREMI @"
2500 PRINT
2600 PRINT"PER COMINCIARE PREMI £"
3000 GETB$:IFB$ <>"£"THEN3000
3500 RETURN
3600 PRINT"■■■■■■QUANTI CARATTERI"
3700 INPUTJ
3710 X=J
3720 DIMA$(X)
3800 RETURN
4000 REM BY COLOMBO C.
5000 REM PINEROLO
5500 REM
6000 REM 16/10/83.
6500 REM
7000 FORT=1TO(2*P):NEXTT:RETURN
8000 PRINT"■"
8500 POKE36879,25
9000 PRINT:PRINT:PRINT:PRINT:PRINT
9500 PRINT"ALLA PROSSIMA V0LTA"
9800 PRINT:PRINT:
9900 PRINT"CIAO"
```

In aria col computer

Il secondo programma provvede alla conversione istantanea delle lettere battute sulla tastiera in codice morse.

Quindi, con una semplice interfaccia saremo pronti a nuovi e interessantissimi QSO in CW.

Attenzione: è necessario adeguarsi alla velocità di trasmissione, cioè non immettere lettere più velocemente di quanto vengano elaborate, oppure qualcuna verrà dimenticata.

Una piccola modifica alla linea 43 permetterà di allineare orizzontalmente una lettera dopo l'altra. Il programma, infatti, prevede l'incolonnamento verticale.

43 PRINT A $ (K);

Nel frattempo, sto studiando il programma per la decodifica del CW, ma il problema è molto più complesso.

```
1 PRINT"■":GOSUB1000
2 PRINT"■":GOSUB500:PRINT"■":GOTO10
3 POKES2,236:FORT=1TOP:NEXTT:POKES2,0:FORT=1TOP:NEXTT:RETURN
4 POKES2,236:FORT=1TOL:NEXTT:POKES2,0:FORT=1TOP:NEXTT:RETURN
5 FORT=1TO(2*P):NEXTT:GOTO30
10 S2=36876
20 POKE36878,10
25 PRINT"■■■■■■■■■■SCRIVI■"
26 PRINT:PRINT
30 GETA$:IFA$=""THEN30
32 POKE36879,28
35 PRINT TAB(5)A$
40 IFA$="A"THENGOSUB3:GOSUB4:GOTO5
50 IFA$="B"THENGOSUB4:GOSUB3:GOSUB3:GOSUB3:GOTO5
60 IFA$="C"THENGOSUB4:GOSUB3:GOSUB4:GOSUB3:GOTO5
70 IFA$="D"THENGOSUB4:GOSUB3:GOSUB3:GOTO5
```

```
80 IFA$="E"THENGOSUB3:GOTO5
90 IFA$="F"THENGOSUB3:GOSUB3:GOSUB4:GOSUB3:GOTO5
100 IFA$="G"THENGOSUB4:GOSUB4:GOSUB3:GOTO5
110 IFA$="H"THENGOSUB3:GOSUB3:GOSUB3:GOSUB3:GOTO5
120 IFA$="I"THENGOSUB3:GOSUB3:GOTO5
130 IFA$="J"THENGOSUB3:   GOSUB4:GOSUB4:GOSUB4:GOTO5
140 IFA$="K"THENGOSUB4:GOSUB3:GOSUB4:GOTO5
150 IFA$="L"THENGOSUB3:GOSUB4:GOSUB3:GOSUB3:GOTO5
160 IFA$="M"THENGOSUB4:GOSUB4:GOTO5
170 IFA$="N"THENGOSUB4:GOSUB3:GOTO5
180 IFA$="O"THENGOSUB4:GOSUB4:GOSUB4:GOTO5
190 IFA$="P"THENGOSUB3:GOSUB4:GOSUB4:GOSUB3:GOTO5
200 IFA$="Q"THENGOSUB4:GOSUB4:GOSUB3:GOSUB4:GOSUB5
210 IFA$="R"THENGOSUB3:GOSUB4:GOSUB3:GOTO5
220 IFA$="S"THENGOSUB3:GOSUB3:GOSUB3:GOTO5
230 IFA$="T"THENGOSUB4:GOTO5
240 IFA$="U"THENGOSUB3:GOSUB3:GOSUB4:GOTO5
250 IFA$="V"THENGOSUB3:GOSUB3:GOSUB3:GOSUB4:GOTO5
260 IFA$="W"THENGOSUB3:GOSUB4:GOSUB4:GOTO5
270 IFA$="X"THENGOSUB4:GOSUB3:GOSUB3:GOSUB4:GOTO5
280 IFA$="Y"THENGOSUB4:GOSUB3:GOSUB4:GOSUB4:GOTO5
290 IFA$="Z"THENGOSUB4:GOSUB4:GOSUB3:GOSUB3:GOTO5
300 IFA$="1"THENGOSUB3:GOSUB4:GOSUB4:GOSUB4:GOSUB4:GOTO5
310 IFA$="2"THENGOSUB3:GOSUB3:GOSUB4:GOSUB4:GOSUB4:GOTO5
320 IFA$="3"THENGOSUB3:GOSUB3:GOSUB3:GOSUB4:GOSUB4:GOTO5
330 IFA$="4"THENGOSUB3:GOSUB3:GOSUB3:GOSUB3:GOSUB4:GOTO5
340 IFA$="5"THENGOSUB3:GOSUB3:GOSUB3:GOSUB3:GOSUB3:GOTO5
350 IFA$="6"THENGOSUB4:   GOSUB3:GOSUB3:GOSUB3:GOSUB3:GOTO5
360 IFA$="7"THENGOSUB4:GOSUB4:GOSUB3:GOSUB3:GOSUB3:GOTO5
370 IFA$="8"THENGOSUB4:GOSUB4:GOSUB4:GOSUB3:GOSUB3:GOTO5
380 IFA$="9"THENGOSUB4:GOSUB4:GOSUB4:GOSUB4:GOSUB3:GOTO5
390 IFA$="0"THENGOSUB4:GOSUB4:GOSUB4:GOSUB4:GOSUB4:GOTO5
400 IFA$=CHR$(32)THENGOTO450
410 IFA$="*"THENGOTO2
415 IFA$="@"THENGOTO8000
450 FORT=1TO(3*P):NEXTT:GOTO30
500 POKE36879,31
510 PRINT"♥"   :PRINT:PRINT:PRINT:PRINT
550 PRINT"QUANTE BATTUTE",
560 PRINT
570 PRINT"AL MINUTO";
600 INPUT K
650 P=INT(3100)/K
700 L=3*P
750 RETURN
1000 PRINT"****CODIFICATORE****",
1100 PRINT
1200 PRINT"           MORSE"
1250 PRINT:PRINT
1300 PRINT"ISRUZIONI"
1350 PRINT
1400 PRINT"IL COMPIUTER PUO"
1500 PRINT"TENERE IN MEMORIA"
```

```
1600 PRINT"SOLO 15 LETTERE PER"
1700 PRINT"VOLTA. ADEGUATI ALLA"
1800 PRINT"VELOCITA' DI TRSMIS-"
1900 PRINT"SIONE"
2000 PRINT:PRINT
2100 PRINT"PER VARIARE LA VELO-"
2200 PRINT"CITA' PREMI *"
2300 PRINT
2400 PRINT"PER FINIRE PREMI @"
2500 PRINT
2600 PRINT"PER COMINCIARE PREMI £"
3000 GETB$:IFB$ <>"£"THEN3000
3500 RETURN
4000 REM BY COLOMBO C.
5000 REM PINEROLO
5500 REM
6000 REM 10/10/83.
6500 REM
8000 PRINT"J"
8500 POKE36879,25
9000 PRINT:PRINT:PRINT:PRINT:PRINT
9500 PRINT"ALLA PROSSIMA V0LTA"
9800 PRINT:PRINT:
9900 PRINT"CIAO"
```

FINE

Marco Bisaccioni

Utility vengono chiamati quei programmi che ci danno un aiuto particolare in determinate situazioni di softwarelavoro. È il caso di questo programmino che, corto corto all'aspetto, è di una UTYLITÀ sorprendente.

LIST & SCROLL PER C64

Le sue caratteristiche peculiari sono:
1 - Genera un listato più facilmente comprensibile incolonnando ciascuna istruzione, all'interno della stessa linea, in una sola esclusiva linea di programma.

Esempio: il seguente programma basic:

```
10 A=10:FORJ=1TO4:FORI
=0TO10:PRINTI;:PRINTA*
B:NEXTI:PRINT"PASSO"J"
OK":B=A+B:NEXTJ:IFJ=AT
HENA=B:GOTO5:END
```

Diventa dopo l'uso di **"List & Scroll"**:

```
10
  A=10:
  FORJ=1TO4:
    FORI=0TO10:
      PRINTI::
      PRINTA*B:
      NEXTI:
    PRINT"PASSO"J"OK":
    B=A+B:
    NEXTJ:
  IFJ=ATHENA=B:
  GOTO5:
  END
```

2 - Permette lo scroll completo del listato in entrambe le direzioni verticali.

Quando avrete finito di ricopiare il listato a piè pagina, controllate accuratamente quanto battuto e poi procedete al Run.

Da questo momento in poi avrete un comando in più sul vostro C-64:
(.) Con la pressione del punto seguito da return avete ora un listato del vostro programma in memoria sicuramente più leggibile e comprensibile.

I tasti 'UP' e 'DOWN', dopo il return, vi permettono lo scroll in avanti e indietro del listato.

Logica conseguenza è il fatto che se al (.) segue la linea di programma, il listato sarà visualizzato da quella particolare linea richiesta:
Ad esempio: . 10 (return)
listerà la linea 10.

Il comando basic LIST esiste ancora ed è a nostra completa disposizione pronto per essere casomai utilizzato alternativamente al **(.) "Lista e Scrolla"**.

```
1 REM##############################################################
2 REM#                 L I S T A     E     S C R O L L A            #
3 REM##############################################################
10 HB=PEEK(56)*256+PEEK(55)-399:I=6769
20 READA:IFA=256THEN40
30 POKEI,A:C=C+A:I=I+1:GOTO20
40 IFC<>51322THENPRINT"ERRORE NEI DATA":END
50 LB=6769
80 READA
100 FORI=0TO382
103 POKEHB+I,PEEK(LB+I)
105 IFA<>LB+IGOTO170
110 V=PEEK(A)+PEEK(A+1)*256:A=V+HB-LB
120 POKEHB+I,A-INT(A/256)*256:I=I+1:POKEHB+I,INT(A/256):READA
170 NEXT
180 SYSHB+2:END
6769 DATA113,26,173,113,26,133,55
6777 DATA133,51,173,114,26,133,56
6785 DATA133,52,234,234,234,169,76
6793 DATA133,124,173,147,26,133,125
6801 DATA173,148,26,133,126,96,149
6809 DATA26,201,46,208,9,72,173
6817 DATA122,0,201,0,240,9,104
6825 DATA201,58,144,1,96,76,128
```

```
6833 DATA0,169,2,141,251,0,32
6841 DATA115,0,240,14,176,21,32
6849 DATA107,169,32,209,26,32,215
6857 DATA170,76,42,165,169,0,133
6865 DATA20,133,21,24,144,238,76
6873 DATA8,175,234,234,234,32,19
6881 DATA166,160,2,177,95,133,20
6889 DATA200,177,95,133,21,160,0
6897 DATA177,95,201,0,208,47,200
6905 DATA177,95,201,0,208,40,240
6913 DATA69,169,0,197,20,208,6
6921 DATA197,21,240,59,198,21,198
6929 DATA20,32,19,166,160,2,177
6937 DATA95,197,20,208,231,200,177
6945 DATA95,197,21,208,224,32,68
6953 DATA229,24,144,201,32,93,27
6961 DATA32,228,255,201,0,240,249
6969 DATA201,145,240,204,201,80,234
6977 DATA234,234,201,13,240,8,230
6985 DATA20,208,160,230,21,208,156
6993 DATA96,56,233,127,170,132,73
7001 DATA160,255,202,240,8,200,185
7009 DATA158,160,16,250,48,245,200
7017 DATA185,158,160,48,6,32,210
7025 DATA255,208,245,96,164,73,41
7033 DATA127,32,210,255,96,160,2
7041 DATA32,215,170,230,199,177,95
7049 DATA170,200,177,95,32,205,189
7057 DATA198,199,32,215,170,166,251
7065 DATA32,228,27,169,0,133,253
7073 DATA160,3,200,177,95,201,0
7081 DATA240,83,166,253,208,4,201
7089 DATA128,176,27,32,210,255,201
7097 DATA34,208,8,72,165,253,73
7105 DATA1,133,253,104,201,58,240
7113 DATA38,208,220,234,234,234,234
7121 DATA234,234,201,130,208,6,206
7129 DATA251,0,206,251,0,72,32
7137 DATA54,27,104,201,129,240,36
7145 DATA201,167,208,191,230,252,230
7153 DATA252,24,144,184,32,215,170
7161 DATA169,0,133,253,165,251,101
7169 DATA252,170,32,228,27,24,144
7177 DATA166,169,0,133,252,133,253
7185 DATA96,230,251,230,251,208,153
7193 DATA224,0,240,7,32,63,171
7201 DATA202,24,144,245,96,256
7700 DATA6769,6772,6779,6793,6798,6803
7705 DATA6842,6935,7029,7091,7122,0

READY.
```

Sperando di avervi ragguagliato esaurientemente, scrollo da questa pagina: END.

FINE

ing. Remo Petritoli

Nella costruzione di apparecchiature a microprocessore capita di frequente di usare delle eprom per memorizzare programmi.
Si tratta indubbiamente di un componente adattissimo allo scopo dato che può esser facilmente programmato e con altrettanta facilità cancellato e riutilizzato.

RUZIONE DI UN ULATORE DI EPROM

Non è tuttavia molto comodo il suo impiego nel corso dello sviluppo di un programma in quanto in tale fase sono frequenti gli errori e le correzioni; oguna di queste richiede purtroppo lo smontaggio della eprom, la sua cancellazione con la lampada a ultravioletti, e una nuova programmazione. Dato che la cosa si ripete molte volte nel corso del lavoro, la perdita di tempo che ne deriva è sensibile. In più i ripetuti montaggi e smontaggi dello zoccolo del prototipo finiscono col danneggiare i contatti; d'altra parte uno zoccolo textol (a forza di inserzione zero) pare sprecato per un uso temporaneo.
Così mi sono deciso a realizzare un circuitino molto comodo che evita tutti gli inconvenienti di cui sopra; lo chiameremo "Emulatore di eprom".

Prima di procedere alla sua descrizione sarà il caso di illustrare la mia situazione e le mie esigenze e di definire alcuni termini con la speranza di rendere più chiara la spiegazione sul funzionamento e l'uso del circuito. Io mi occupo attualmente della costruzione di piccoli controller a microprocessor. Per ora uso lo Z80 con una eprom, una ram e circuiti periferici (PIO,CTC,SIO) adatti alla applicazione. Ne vengono fuori piccoli moduli relativamente compatti ed economici. Sarebbe meglio far uso di computer più adatti allo scopo ma la mancanza di un cross-assembler mi fa rimandare la cosa.

Il lavoro è limitato e non giustifica l'acquisto di un sistema di sviluppo commerciale. Dispongo però di due calcolatori, uno Sharp MZ80K e un Osborne 1 con relativi programmi editore-assembler.

I primi tempi ho provato a metter su eprom i programmi man mano che li sviluppavo per la prova. Ho dovuto subito rinunciare in quanto passavo più tempo a trafficare sulle eprom che a far programmi. Dopo aver provato altre soluzioni mi son deciso a costruire l'emulatore prendendo spunto da un articolo sulla rivista americana BYTE (n. 2/82 pagina 194). Nella mia realizzazione ho fatto in modo che potesse esser collegato con facilità a calcolatori diversi e che si adattasse alle mie necessità.

Quasi tutti i computer hanno una o due porte di entrata e uscita, né è difficile aggiungercene una se manca (è quello che ho fatto io allo Sharp). Anche un modesto ZX80 può così, con un programmino in basic, caricare programmi in codice macchina su piccole schede per controller o per qualunque altra applicazione in cui si usi una 2716.

Questa soluzione, anche se funziona, resta alquanto primitiva e va bene solo per uso hobbistico. Io ho modificato l'assem-

bler in modo da inviare direttamente i bytes alla ram dell'emulatore. In questo modo lo studio e la prova dei programmi sono diventati un lavoretto molto spedito.

Per brevità chiamerò "master" (principale) il computer principale su cui viene sviluppato il programma e che provvede a caricarlo sulla ram dell'emulatore e "slave" (schiavo) il circuito in prova che contiene la 2716 su cui va messo il programma in fase di sviluppo.

Nello slave viene inserito nello zoccolo della eprom uno spinotto a 24 contatti da cui partono altrettanti fili verso l'emulatore. I fili sono brevi in modo da non dar problemi di funzionamento: nel mio caso solo 20 cm.

L'emulatore comprende una ram da 2K (per la 2716) e tutta una serie di buffer per il controllo del flusso dei dati. Il principio di funzionamento è il seguente: nel corso del funzionamento normale la ram è collegata al bus dello slave e viene da questi letta come se fosse una normale 2716.

Per cambiare il programma in questa pseudo-eprom non occorre far altro che scrivere nuovi dati sulle ram. A questo provvede il master.

Vediamo le cose sullo schema a blocchi. La 2716 ha 11 linee di entrata per gli indirizzi e otto linee di uscita per i dati. Queste linee vanno collegate (alternativamente perché non vi siano conflitti sul bus) o allo slave o al master. Vediamo prima le cose quando il circuito emula una eprom. Possiamo supporre allora non collegati Q_5, Q_6, Q_7 (vedi dopo) e considerare solo Q_1, Q_2, Q_3. Questi integrati sono dei buffer a otto bit abilitabili ognuno separatamente in due gruppi di quattro bit. Q_3 invia alla ram le 8 linee di indirizzo da A0 a A7 e Q_{2B} altre tre linee (A8,A9,A10). È ancora disponibile una linea per espandere l'emulatore in modo da poter simulare la 2732.

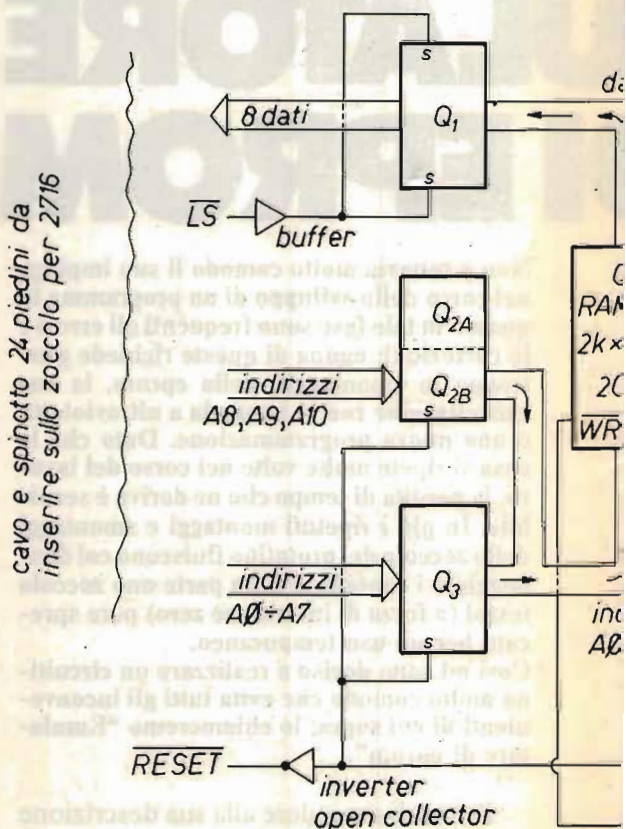
Quando la eprom dello slave viene selezionata dal relativo decoder (piedino 18 della 2716) l'entrata CS va a zero abilitando la ram Q_4 a fornire i dati (piedino 20 RD). Viene abilitato pure Q_1 che li trasmette al bus dello slave. L'effetto complessivo è che questo non vede in alcun modo la differenza tra l'emulatore e una vera 2716 sullo zoccolo.

Possiamo quindi far tutte le prove che vogliamo senza alcun problema. Ma appena scopriamo un errore o una modifica da fare al programma le cose cambiano radicalmente. Basta cambiare lo stato del bit 4 della porta 2 del master per disabilitare Q_1, Q_{2B}, Q_3 e abilitare al loro posto Q_5, Q_6, Q_7. Questi sono dei latch a otto bit (qualcosa di simile ai 7475, ma doppi) con uscita tristato. Le entrate sono collegate all'uscita della porta A del computer. I comandi di memorizzazione sono controllati dai bit 0,1,2 della porta B.

Per scrivere i nuovi dati sulla ram dovremo per ogni byte ripetere la seguente frequenza:

1 - mettere in uscita sulla porta A gli otto bit inferiori dell'indirizzo;

2 - portare a 1 e poi a 0 il bit 0 della porta B per memorizzare il dato in Q_7;

3 - mettere in uscita sulla porta A gli 8 bit

superiori dell'indirizzo (quelli di troppo sono scartati dal circuito);

4 - portare a 1 e poi a 0 il bit 1 della porta B;

5 - mettere in uscita sulla porta A il byte che va scritto in ram;

6 - portare a 1 e poi a 0 il bit 2 della porta B;

7 - portare a 0 e poi a 1 il bit 3 della porta B per scrivere il dato nella ram.

La sequenza può sembrar lunga ma il master la fa con velocità e precisione...

Alla fine della sequenza di programmazione si lascia il bit 3 della porta B a 1 (così la ram è protetta da ulteriori scritture) e si porta a 1 il bit 4 della porta B (disabilitando il circuito di scrittura e mettendo in funzione i buffer che collegano la ram allo zoccolo della 2716 del micro da provare.

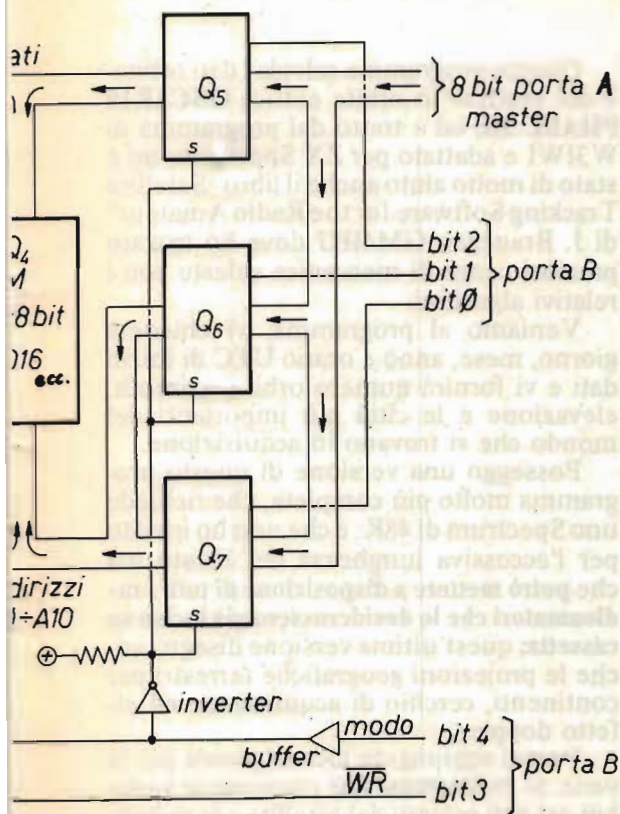
Basta premere il reset per poter provare il nuovo programma. Io ho previsto una uscita TTL open collector che se collegata al reset consente di far il tutto automaticamente: in fase di scrittura sulla ram detta uscita va a zero tenendo resettato lo slave fino a che non si riporta a 1 il bit 4 della porta B. A questo punto la ram è collegata allo slave e il programma parte regolarmente avendo tolto il reset.

Dato che non è usata per gli indirizzi, Q_{2A} viene lasciato sempre abilitato e serve da buffer per alcuni segnali (CS dello slave e il bit 4 della porta B del master).

COSTRUZIONE

Dato il gran numero di collegamenti da effettuare non è il caso di pensare a un circuito stampato. Si possono effettuare i collegamenti con tecnica wire-wrap, ma si può benissimo far uso di pezzetti di filo saldati con molta pazienza sugli zoccoli. Adottare le solite precauzioni necessarie per i TTL, io non ho avuto problemi di sorta. Se questo suggerimento avrà suscitato il vostro interesse, in un successivo articolo proporrò lo schema elettrico completo e un esempio di programma di gestione per l'emulatore.

Spero di essere stato chiaro, resto comunque a disposizione per chiarimenti telefonici sull'articolo (ore 20÷22,30 di tutti i giorni 0736/65880 o 085/292251).

FINE

Q_1, Q_2, Q_3 74LS244
Q_5, Q_6, Q_7 74LS373

IW4ARD, Gianni Matteini

**Ho notato con piacere sulla nostra CQ l'inizio delle rubriche "Io e il computer" e "Radioamatori e Computer".
Sono molto interessato ai programmi per ZX Spectrum ed è per questo che vorrei dare anche io il mio piccolo contributo: credo di far cosa gradita a tutti i radioamatori che hanno lo Spectrum.**

CALCOLO DATI ORBITALI

```
  10 REM PHASE 3B SATELLITE IN O
RBITA ELLITTICA AZIMUTH E ELEVAZ
IONE
  20 POKE 23658,8: GO TO 0820
  30 LET DIF=ABS (W5-LO)
  40 IF DIF>180 THEN LET DIF=360
-DIF
  50 LET DZ1=((SIN (ALT*P0))*(SI
N (L5*P0))+(COS (ALT*P0))*(COS (
L5*P0))*(COS (DIF*P0)))
  60 LET DZ1=57.3*(-ATN (DZ1/SQR
 (-DZ1*DZ1+1))+P1/2)
  70 IF DZ>=DZ1 THEN PRINT PAPER
 4;AT N,0;"IN ";P$
  80 IF DZ<DZ1 THEN PRINT PAPER
2;AT N,0;" . ";P$
  90 RETURN
 100 LET SLO=46-INT (LO/12+.5)
 110 IF LO<180 THEN LET SLO=15-I
NT (LO/12-.5)
 120 LET SLA=INT (12.5-ALT/8)
 130 IF DZ>DZ1 THEN PAPER 2: PRI
NT AT SLA,SLO;"*"
 140 RETURN
 150 REM
 160 REM PIANO DX
 170 REM
 180 LET ALT=51.5
 190 LET LO=0
 200 LET N=4: LET P$="LONDON"
 210 GO SUB 0030
 220 LET ALT=40.75
 230 LET LO=74
 240 LET N=5: LET P$="NEW YORK"
 250 GO SUB 0030
 250 LET ALT=37.58
 270 LET LO=122.5
 280 LET N=6: LET P$="S FRANCISC
O"
 290 GO SUB 0030
 300 LET ALT=-23
 310 LET LO=43.2
 320 LET N=7: LET P$="RIO"
 330 GO SUB 0030
 340 LET ALT=-33.92
 350 LET LO=341.63
 360 LET N=8: LET P$="CAPETOWN"
```

Questo programma calcola i dati orbitali del satellite in orbita elittica **OSCAR10 PHASE 3B,** ed è tratto dal programma di W3IWI e adattato per ZX Spectrum; mi è stato di molto aiuto anche il libro "Satellite Tracking Software for the Radio Amateur" di J. Branegan GM4IHJ dove ho trovato preziosi cenni di meccanica celeste con i relativi algoritmi.

Veniamo al programma: vi chiederà giorno, mese, anno e orario UTC di inizio dati e vi fornirà numero orbita, azimuth, elevazione e le città più importanti del mondo che si trovano in acquisizione.

Possego una versione di questo programma molto più completa, che richiede uno Spectrum di 48K, e che non ho inviato per l'eccessiva lunghezza del listato ma che **potrò mettere a disposizione di tutti i radioamatori che lo desiderassero già inciso su cassetta;** quest'ultima versione disegna anche le proiezioni geografiche terrestri per continenti, cerchio di acquisizione ed effetto doppler.

Vorrei aggiungere alcune parole per la parte di programma che contiene le variabili dei dati orbitali del satellite e le coordinate geografiche della propria posizione.

```
370 GO SUB 0030
380 LET ALT=1.3
390 LET LO=256.15
400 LET N=9: LET P$="SINGAPORE"
410 GO SUB 0030
420 LET ALT=22.2
430 LET LO=245.75
440 LET N=10: LET P$="HONG KONG
"
450 GO SUB 0030
460 LET ALT=35.75
470 LET LO=220.25
480 LET N=11: LET P$="TOKYO"
490 GO SUB 0030
500 LET ALT=-12.33
510 LET LO=229.1
520 LET N=12: LET P$="DARWIN"
530 GO SUB 0030
540 LET ALT=-31.95:
550 LET LO=244.13
560 LET N=13: LET P$="PERTH"
570 GO SUB 0030
580 LET ALT=-33.88
590 LET LO=208.83
600 LET N=14: LET P$="SYDNEY"
610 GO SUB 0030
620 LET ALT=-41.317
630 LET LO=165.23
640 LET N=15: LET P$="WELLINGTO
N"
650 GO SUB 0030
660 LET ALT=21.313
670 LET LO=157.87
680 LET N=16: LET P$="HAWAII"
690 GO SUB 0030
700 LET ALT=65
710 LET LO=147.66
720 LET N=17: LET P$="FAIRBANKS
"
730 GO SUB 0030
740 RETURN
750 LET D$=STR$ (T4)
760 DIM E$(4)
770 LET E$="0000"
780 DIM T$(4)
790 LET U=4-LEN D$
800 LET T$=E$(1 TO U)+D$
810 RETURN
820 BORDER 1: PAPER 1: INK 7: C
LS
830 PRINT INK 2; PAPER 6; FLASH
 1;"PHASE 3 CALCOLO DATI ORBITAL
I ": PRINT
840 PRINT "INSERIRE NUMERO GIOR
NO"
850 INPUT DM
860 PRINT INVERSE 1;AT 2,28;DM:
PRINT
870 PRINT "INSERIRE NUMERO MESE
"
880 INPUT MN
890 PRINT INVERSE 1;AT 4,28;MN:
PRINT
900 PRINT "INSERIRE NUMERO ANNO
ES.83"
910 INPUT YN
920 PRINT INVERSE 1;AT 6,28;YN:
PRINT
930 PRINT ,,
940 PRINT "INSERIRE ORARIO UTC,
DI PARTENZA DEI DATI UTC IN HHMM
"
950 INPUT A$
960 PRINT AT 21,21;"HHMM "; INV
ERSE 1;A$
970 LET T1=VAL A$
980 FOR N=0 TO 400: NEXT N
990 LET D6=INT (YN/4)-INT ((YN-
1)/4)+212+DM+(SGN (MN-8)*(INT (.
5+(ABS ((MN-8)*30.5)))))
1000 IF MN<3 THEN LET D6=DM+(MN-
1)*31
1010 CLS
1020 GO SUB 2420
1030 LET G0=7.5369793E13
1040 LET G1=1.0027379093
1050 LET G2=.2759328721
1060 IF YN=83 THEN LET G2=.27526
97531
1070 IF YN=84 THEN LET G2=.27460
65342
1080 LET D6=D6+365*(YN-Y2)
1090 LET P1=3.14159265
1100 LET P2=2*P1
1110 LET P0=P1/180
1120 LET T2=(INT (T1/100))/24+(T
1-(INT (T1/100))*100)/(60*24)+D6
1130 LET T=T2
1140 LET CH4=5
1150 LET R0=6378.16
1160 LET F=1/298.25
1170 IF N0>.1 THEN LET A0=((G0/(
N0↑2))↑(1/3))
1180 IF N0<=.1 THEN LET N0=SQR (
G0/(A0↑3))
1190 LET E2=1-E0↑2
1200 LET E1=SQR (E2)
1210 LET FA=0
1220 LET Q0=M0/360+K0
1230 LET Q=N0*(T-T0)+Q0
1240 LET K=INT Q
1250 BORDER 0: PAPER 0: INK 7
1260 CLS
1270 PRINT PAPER 4; INK 0;AT 0,6
;"PHASE 3B AL ";DM;"/";MN;"/";YN
: PRINT
1280 PRINT INVERSE 1;AT 1,19,"OR
BIT N. ";K
1290 PRINT PAPER 2; INK 6;AT 2,1
9;"UTC";TAB 25;"AZ";TAB 29;"EL"
1300 PRINT TAB 19;"-------------"
1310 LET K2=9.95*((R0/A0)↑3.5)/(
E2↑2)
1320 LET S1=SIN (I0*P0)
1330 LET C1=COS (I0*P0)
1340 LET O=O0-(T-T0)*K2*C1
1350 LET S0=SIN (O*P0)
1360 LET C0=COS (O*P0)
1370 LET W=W0+(T-T0)*K2*(2.5*(C1
↑2)-.5)
1380 LET S2=SIN (W*P0)
1390 LET C2=COS (W*P0)
1400 DIM C(3,2)
1410 LET C(1,1)=(C2*C0)-(S2*S0*C
1)
1420 LET C(1,2)=-(S2*C0)-(C2*S0*
C1)
1430 LET C(2,1)=(C2*S0)+(S2*C0*C
1)
1440 LET C(2,2)=-(S2*S0)+(C2*C0*
C1)
1450 LET C(3,1)=(S2*S1)
1460 LET C(3,2)=(C2*S1)
1470 LET Q=N0*(T-T0)+Q0
1480 LET K=INT Q
1490 LET M9=INT ((Q-K)*256)
1500 LET M=(Q-K)*P2
1510 LET E=M+E0*SIN M+.5*(E0↑2)*
SIN (2*M)
1520 LET S3=SIN E
1530 LET C3=COS E
1540 LET R3=1-E0*C3
1550 LET M1=E-E0*S3
1560 LET M5=M1-M
1570 IF ABS (M5)<1E-6 THEN GO TO
 1600
1580 LET E=E-M5/R3
1590 GO TO 1520
1600 LET X0=A0*(C3-E0)
1610 LET Y0=A0*E1*S3
1620 LET R=A0*R3
1630 LET X1=X0*C(1,1)+Y0*C(1,2)
1640 LET Y1=X0*C(2,1)+Y0*C(2,2)
1650 LET Z1=X0*C(3,1)+Y0*C(3,2)
1660 LET G7=T*G1+G2
1670 LET G7=(G7-(INT G7))*P2
1680 LET S7=-SIN G7
1690 LET C7=COS G7
1700 LET X=(X1*C7)-(Y1*S7)
1710 LET Y=(X1*S7)+(Y1*C7)
1720 LET Z=Z1
1730 LET L8=L9*P0
1740 LET S9=SIN (L8)
1750 LET C9=COS (L8)
1760 LET S8=SIN (-W9*P0)
1770 LET C8=COS (W9*P0)
1780 LET R9=R0*(1-(F/2)+(F/2)*CO
S (2*L8))+H9/1000
1790 LET L8=ATN ((1-F)↑2*S9/C9)
1800 LET Z9=R9*SIN (L8)
1810 LET X9=R9*COS (L8)*C8
1820 LET Y9=R9*COS (L8)*S8
1830 LET X5=(X-X9)
1840 LET Y5=(Y-Y9)
```

```
1850 LET Z5=(Z-Z9)
1860 LET R5=SQR (X5*X5+Y5*Y5+Z5*
Z5)
1870 LET DZ=6378/R
1880 LET DZ=57.3*(-ATN (DZ/SQR (
-DZ*DZ+1))+P1/2)
1890 LET Z8=(X5*C8*C9)+(Y5*S8*C9
)+(Z5*S9)
1900 LET X8=-(X5*C8*S9)-(Y5*S8*S
9)+(Z5*C9)
1910 LET Y8=(Y5*C8)-(X5*S8)
1920 LET S5=Z8/R5
1930 LET C5=SQR (1-S5*S5)
1940 LET E9=(ATN (S5/C5))/P0
1950 IF E9<-5 THEN GO TO 2300
1960 LET A9=(ATN (Y8/X8))/P0
1970 LET B5=Z/R
1980 LET L5=(ATN (B5/(SQR (1-B5*
B5))))*57.3
1990 LET W5=(ATN (Y/X))*57.3
2000 IF X<0 THEN LET W5=180-W5
2010 IF X>0 AND Y<0 THEN LET W5=
-W5
2020 IF X>0 AND Y>0 THEN LET W5=
360-W5
2030 IF X=0 AND Y>=0 THEN LET W5
=270
2040 IF X=0 AND Y<0 THEN LET W5=
90
2050 IF X8<0 THEN LET A9=A9+180
2060 IF X8>0 AND Y8<0 THEN LET A
9=360+A9
2070 IF X8=0 AND Y8>=0 THEN LET
A9=90
2080 IF X8=0 AND Y8<0 THEN LET A
9=270
2090 LET T4=(INT ((T-INT T)*2400
+.5))/100
2100 LET T4=INT ((100*((T4-INT T
4)*.6+INT T4))+.5)
2110 GO SUB 0750
2120 IF FA=1 THEN GO TO 2210
2130 PRINT TAB 19;T$;TAB 25;INT
A9;TAB 29;INT E9;,,
2140 LET T=T+1/96
2150 IF (T-T2)>.16 THEN GO TO 21
70
2160 GO TO 1340
2170 LET FA=1
2180 LET T=T2
2190 PRINT AT 20,0; PAPER 1; INK
 7;"TENERE PREMUTO: L PER UN NUO
VO MENU, A PER FERMARE, C PER S
TOP "
2200 GO TO 1340
2210 PRINT AT 2,0;"DX STATE AT "
;T$
2220 PRINT AT 3,0;"-------------
---"
2230 GO SUB 0160
2240 IF INKEY$="A" THEN GO TO 22
70
2250 IF INKEY$<>"A" THEN GO TO 2
300
2260 IF INKEY$="C" THEN GO TO 23
40
2270 PRINT AT 18,9;"FERMO"
2280 GO TO 2260
2290 PRINT AT 18,9;"     "
2300 IF INKEY$="L" THEN BORDER 1
: PAPER 1: CLS : GO TO 930
2310 LET T=T+1/96
2320 IF (T-T2)>.16 THEN GO TO 21
70
2330 GO TO 1340
2340 CLS
2350 PRINT "TITOLO........PHASE
3B"
2360 PRINT "IW4ARD Gianni MATTEI
NI"
2370 PRINT "Via C. Pavese, 20"
2380 PRINT "47041 BELLARIA (Forl
i')"
2390 PRINT "Adattato dal program
ma di W3IWI"
2400 PRINT "Per ZX Spectrum."
2410 STOP
2420 REM INSERIRE YEAR NUMBER
OF ORBIT DATA EPOCH
2430 LET Y2=83
2440 REM INSERIRE DAY.DAY OF
ORBIT DATA EPOCH
2450 LET T0=285.5
2460 REM INSERIRE ORBIT INCLINAT
ION ANGLE IN DEGREES.DEGREES
2470 LET I0=26.0050
2480 REM INSERIRE ORBIT REVOLUTI
ON COUNT AT EPOCH TIME. INSERIRE
 0 SE SCONOSCIUTO
2490 LET K0=249
2500 REM INSERIRE MEAN ANOMALY
AT EPOCH IN DEG.DEG
2510 LET M0=17.2220
2520 REM INSERIRE MEAN MOTION
IN REV.REV PER DAY
2530 LET N0=2.05854010
2540 REM INSERIRE ECCENTRICITY
2550 LET E0=.6049226
2560 REM INSERIRE ARGUMENT OF
PERIGEE IN DEG.DEG
2570 LET W0=210.5700
2580 REM INSERIRE RIGHT ASCENSIO
N OF ASCENDING NODE
2590 LET O0=236.6270
2600 REM INSERIRE PROPRIA LATITU
DINE IN DEG.DEG (SUD LAT. E' -ve
)
2610 LET L9=44.0854
2620 REM INSERIRE PROPRIA LONGIT
UDINE IN DEG.DEG OVEST
2630 LET W9=347.3224
2640 REM INSERIRE ALTEZZA DELLA
STAZIONE SUL LIVELLO DEL MARE
IN METRI
2650 LET H9=20
2660 RETURN
2670 SAVE "PHASE 3B" LINE 1
```

Le linee vanno dalla 2420 alla 2650; alla 2610 e alla 2630 si trovano rispettivamente latitudine e longitudine in gradi, gradi; esempio 44° 30' si scriverà 44.5 etc, la longitudine va inserita come longitudine Ovest esempio 12° 30' diventano 347° 30' che andranno inseriti come 347.5, al passo di programma 2650 va inserita l'altezza in metri sul livello del mare della propria stazione.

Con questo ho terminato!

I dati immessi nel programma sono aggiornati con dati del mese di Ottobre '83 forniti da KA9Q, via AMSAT Net sui 20 m.

Sono disponibile per qualsiasi informazione sul programma:

IW4ARD: MATTEINI Gianni
Via C. Pavese, 20
47041 BELLARIA (FO)
Tel. 0541-44292

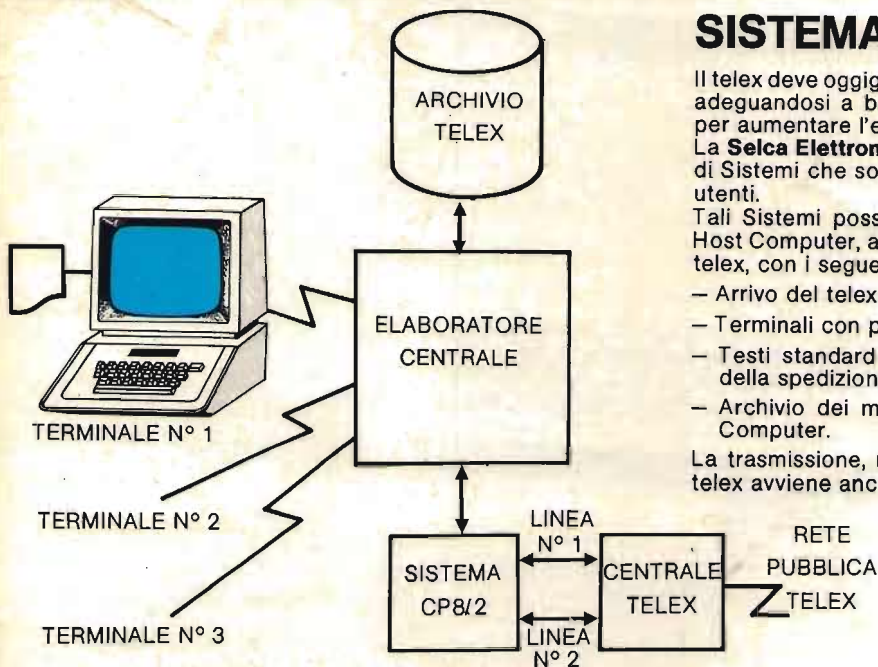
FINE

Omologazione Ministero P.T.
n.: IST 0011119/3-558 del 17-11-1983.

SISTEMA CP8/2-IB

Il telex deve oggigiorno fornire un servizio ottimale, adeguandosi a ben specifiche esigenze d'ufficio per aumentare l'efficienza.
La **Selca Elettronica** propone una serie completa di Sistemi che soddisfano i piccoli, medi e grandi utenti.
Tali Sistemi possono anche essere collegati ad Host Computer, abilitandone i terminali al servizio telex, con i seguenti principali servizi:

- Arrivo del telex al terminale interessato.
- Terminali con pass-word di autorizzazione.
- Testi standard da personalizzare al momento della spedizione.
- Archivio dei messaggi nella memoria di Host Computer.

La trasmissione, ricezione e memorizzazione del telex avviene anche con calcolatore spento.

Sede: Amministrazione e Stabilimento: 40012 CALDERARA di RENO (BO) - Via XXV Aprile 27 - Tel. (051) 727271 (3 linee R.A.) - Telex 216699

Direzione Commerciale Lombardia: 20143 MILANO - Via F. Venosta 31 - Tel. (02) 817595-816457-817533 - Telex 324684

Direzione Commerciale Centro Sud: 00183 ROMA - Via Appia Nuova 59 - Tel. (06) 7553665-7573397 - Telex 620174

SELCA